OSPF Version 2 Traps

Rob Coltun
Consultant
(301) 340-9416
rcoltun@ni.umd.edu

Fred Baker
Advanced Computer Communications
315 Bollay Drive
Santa Barbara, California 93117
fbaker@acc.com

Table Of Contents

1.0 Status Of This Memo ...................................... 2
2.0 Abstract ................................................ 2
3.0 The Network Management Framework ........................ 2
4.0 Objects .................................................. 3
4.0 Format Of Definitions .................................... 3
5.0 Approach ................................................ 4
5.1 Ignoring Initial Activity ................................. 4
5.2 Throttling Traps ........................................ 4
5.3 One Trap Per OSPF Event ................................. 4
5.4 Polling Event Counters ................................ 5
6.0 OSPF Trap Definitions ................................... 6
6.1 Imports .................................................. 6
6.2 Trap Support Objects ................................... 6
6.3 Traps ..................................................... 7
7.0 Acknowledgements ........................................ 12
8.0 References .............................................. 12
1.0 Status Of This Memo

This document is an Internet Draft. 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. Internet Drafts may be updated, replaced, or obsoleted by other documents at any time. It is not appropriate to use Internet Drafts as reference material or to cite them other than as a "working draft" or "work in progress."

Please check the I-D abstract listing contained in each Internet Draft directory to learn the current status of this or any other Internet Draft.

This memo does not, in its draft form, specify a standard for the Internet community.

2.0 Abstract

OSPF [1] is an event driven routing protocol, where an event can be a change in an OSPF interface’s link-level status, the expiration of an OSPF timer or the reception of an OSPF protocol packet. Many of the actions that OSPF takes as a result of these events will result in a change of the routing topology. As routing topologies become large and complex it is often difficult to locate the source of a topology change or unpredicted routing path by polling a large number or routers. Another approach is to notify a network manager of potentially critical OSPF events with SNMP traps.

This draft document defines a set of traps, objects and mechanisms to enhance the ability to manage IP internetworks which use OSPF as its IGP. It is meant as an extension to the OSPF MIB [2].

3.0 The Network Management Framework

The Internet-standard Network Management Framework consists of three components. They are:

RFC 1155 [5] which defines the SMI, the mechanisms used for describing and naming objects for the purpose of management.
RFC 1212 [8] defines a more concise description mechanism, which is wholly consistent with the SMI.

RFC 1157 [7] which defines the SNMP, the protocol used for network access to managed objects.

The Framework permits new objects to be defined for the purpose of experimentation and evaluation.

4.0 Objects

Managed objects are accessed via a virtual information store, termed the Management Information Base or MIB. Objects in the MIB are defined using the subset of Abstract Syntax Notation One (ASN.1) [3] defined in the SMI. In particular, each object has a name, a syntax, and an encoding. The name is an object identifier, an administratively assigned name, which specifies an object type. The object type together with an object instance serves to uniquely identify a specific instantiation of the object. For human convenience, we often use a textual string, termed the OBJECT DESCRIPTOR, to also refer to the object type.

The syntax of an object type defines the abstract data structure corresponding to that object type. The ASN.1 language is used for this purpose. However, the SMI purposely restricts the ASN.1 constructs which may be used. These restrictions are explicitly made for simplicity.

The encoding of an object type is simply how that object type is represented using the object type’s syntax. Implicitly tied to the notion of an object type’s syntax and encoding is how the object type is represented when being transmitted on the network.

The SMI specifies the use of the basic encoding rules of ASN.1 [4], subject to the additional requirements imposed by the SNMP.

The SMI does not provide a means for defining traps. The SNMP, however, does define a few standard traps and provides a means for defining enterprise-specific traps. The TRAP-TYPE macro defined in RFC 1215 [10] suggests a straight-forward approach towards defining traps used with the SNMP.

4.1 Format of Definitions

Section 6 contains the specification of all object types contained in this MIB module. The object types are defined using the conventions defined in the SMI, as amended by the extensions specified in RFC 1212.

Section 6.3 contains the trap definitions using the TRAP-TYPE macro described in RFC 1215.
5.0 Approach

The mechanism for sending traps is straight-forward. When an exception event occurs, the application notifies the local agent who sends a trap to the appropriate SNMP management stations. The message includes the trap type and may include a list of trap specific variables. A new object is defined in section 6.2 that will allow a network manager to enable or disable particular OSPF traps. Section 6.3 gives the trap definitions which includes the variable lists. The router ID of the originator of the trap is included in the variable list so that the network manager may easily determine the source of the trap.

To limit the frequency of OSPF traps, the following additional mechanisms are suggested.

5.1 Ignoring Initial Activity

The majority of critical events occur when OSPF is enabled on a router, at which time the designated router is elected and neighbor adjacencies are formed. During this initial period a potential flood of traps is unnecessary since the events are expected. To avoid unnecessary traps, a router should not originate expected OSPF interface related traps until two of that interface’s dead timer intervals have elapsed. The expected OSPF interface traps are ospfIfStateChange, ospfVirtIfStateChange, ospfNbrStateChange, ospfVirtNbrStateChange, ospfTxRetranmit and ospfVirtIfTxRetransmit. Additionally, ospfMaxAgeLsa and ospfOriginLsa traps should not be originated until two dead timer intervals have elapsed where the dead timer interval used should be the dead timer with the smallest value.

5.2 Throttling Traps

The mechanism for throttling the traps is similar to the mechanism explained in RFC 1224 [11] section 5. The basic idea is that there is a sliding window in seconds and an upper bound on the number of traps that may be generated within this window. Unlike RFC 1224, traps are not sent to inform the network manager that the throttling mechanism has kicked in.

A single window should be used to throttle all OSPF traps types except for the ospfLsdbOverflow and the ospfLsdbApproachingOverflow trap which should not be throttled. For example, if the window time is 3, the upper bound is 3 and the events that would cause trap types 1,3,5 and 7 occur within a 3 second period, the type 7 trap should not be generated.

Appropriate values are 7 traps with a window time of 10 seconds.

5.3 One Trap Per OSPF Event
Several of the traps defined in section 6.3 are generated as the result of finding an unusual condition while parsing an OSPF packet or a processing a timer event. There may be more than one unusual condition detected while handling the event. For example, a link-state update packet may contain several retransmitted link-state advertisements (LSAs), or a retransmitted database description packet may contain several database description entries. To limit the number of traps and variables, OSPF should generate at most one trap per OSPF event. Only the variables associated with the first unusual condition should be included with the trap. Similarly, if more than one type of unusual condition is encountered while parsing the packet, only the first event will generate a trap.

5.4 Polling Event Counters

Many of the tables in the OSPF MIB contain generalized event counters. By enabling the traps defined in this document a network manager can obtain more specific information about these events. A network manager may want to poll these event counters and enable specific OSPF traps when a particular counter starts increasing abnormally.

The following table shows the relationship between the event counters defined in the OSPF MIB and the trap types defined in section 6.3.

<table>
<thead>
<tr>
<th>Counter</th>
<th>Trap Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospfOriginateNewLsas</td>
<td>ospfOriginateLsa</td>
</tr>
<tr>
<td>ospfIfEvents</td>
<td>ospfIfStateChange</td>
</tr>
<tr>
<td></td>
<td>ospfConfigError</td>
</tr>
<tr>
<td></td>
<td>ospfIfAuthFailure</td>
</tr>
<tr>
<td></td>
<td>ospfRxBadPacket</td>
</tr>
<tr>
<td></td>
<td>ospfTxRetransmit</td>
</tr>
<tr>
<td>ospfVirtIfEvents</td>
<td>ospfVirtIfStateChange</td>
</tr>
<tr>
<td></td>
<td>ospfVirtIfConfigError</td>
</tr>
<tr>
<td></td>
<td>ospfVirtIfAuthFailure</td>
</tr>
<tr>
<td></td>
<td>ospfVirtIfRxBadPacket</td>
</tr>
<tr>
<td></td>
<td>ospfVirtIfTxRetransmit</td>
</tr>
<tr>
<td>ospfNbrEvents</td>
<td>ospfNbrStateChange</td>
</tr>
<tr>
<td>ospfVirtNbrEvents</td>
<td>ospfVirtNbrStateChange</td>
</tr>
<tr>
<td>ospfExternLSACount</td>
<td>ospfLsdbApproachingOverflow</td>
</tr>
<tr>
<td>ospfExternLSACount</td>
<td>ospfLsdbOverflow</td>
</tr>
</tbody>
</table>
6.0 OSPF Trap Definitions

6.1 Imports

RFCXXX-MIB DEFINITIONS ::= BEGIN
  IMPORTS
    IpAddress, OSPF
    FROM RFC1155-SMI
  OBJECT-TYPE
    FROM RFC1212
  TRAP-TYPE
    FROM RFC1215
  ospfRouterId, ospfIfIpAddress, ospfAddressLessIf,
  ospfAreaId, ospfIfType, ospfIfState, ospfVirtAreaId,
  ospfVirtIfNeighbor, ospfVirtIfState, ospfNbrIpAddr,
  ospfNbrAddressLessIndex, ospfNbrRtrId, ospfNbrState,
  ospfVirtNbrArea, ospfVirtNbrRtrId, ospfLsdbAreaId,
  ospfLsdbType, ospfLsdbLsid, ospfLsdbRouterId,
  ospfExtLsdbLimit
    FROM RFC1253-OSPF MIB

6.2 Trap Support Objects

-- Trap Support Objects

  ospfTrapGroup OBJECT IDENTIFIER ::= { ospf 14 }

  ospfSetTrap OBJECT-TYPE
    SYNTAX OCTET STRING
    ACCESS read-write
    STATUS mandatory
    DESCRIPTION
      "A four-octet string serving as a bit map for the trap events defined by the OSPF traps. This object is used to enable and disable specific OSPF traps where a 1 in the bit field represents enabled. The right-most bit (least significant) represents trap 0."
    ::= { ospfTrapGroup 1 }

  ospfConfigErrorType OBJECT-TYPE
    SYNTAX INTEGER {
      badVersion (1),
      areaMismatch (2),
      unknownNbmaNbr (3), -- Router is Dr eligible
      unknownVirtualNbr (4),
      authTypeMismatch(5),
      authFailure (6),
      netMaskMismatch (7),
      helloIntervalMismatch (8),
      deadIntervalMismatch (9),
      optionMismatch (10) }

ACCESS  read-only
STATUS  mandatory
DESCRIPTION
"Potential types of configuration conflicts. Used
by the ospfConfigError and ospfConfigVirtError
traps."
::= { ospfTrapGroup 2 }

ospfPacketType  OBJECT-TYPE
SYNTAX  INTEGER {
  hello (1),
  dbDescr (2),
  lsReq (3),
  lsUpdate (4),
  lsAck (5) }
ACCESS  read-only
STATUS  mandatory
DESCRIPTION
"OSPF packet types."
::= { ospfTrapGroup 3 }

6.3  Traps

-- Traps

ospfIfStateChange       TRAP-TYPE
ENTERPRISE      ospf
VARIABLES {  ospfRouterId, -- The originator of the trap
              ospfIfIpAddress,
              ospfAddressLessIf,
              ospfIfState } -- The new state
DESCRIPTION
"An ospfIfStateChange trap signifies that there has
been a change in the state of a non-virtual OSPF inter-
face. 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)."
::= 0

ospfVirtIfStateChange   TRAP-TYPE
ENTERPRISE      ospf
VARIABLES {  ospfRouterId, -- The originator of the trap
              ospfVirtAreaId,
              ospfVirtIfNeighbor,
              ospfVirtIfState } -- The new state
DESCRIPTION
"An ospfVirtIfStateChange 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)."

::= 1

ospfNbrStateChange TRAP-TYPE
ENTERPRISE ospf
VARIABLES {
  ospfRouterId, -- The originator of the trap
  ospfNbrIpAddr,
  ospfNbrAddressLessIndex,
  ospfNbrRtrId,
  ospfNbrState } -- The new state
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 an 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."

::= 2

ospfVirtNbrStateChange TRAP-TYPE
ENTERPRISE ospf
VARIABLES {
  ospfRouterId, -- The originator of the trap
  ospfVirtNbrArea,
  ospfVirtNbrRtrId,
  ospfVirtNbrState } -- The new state
DESCRIPTION
"An ospfVirtNbrStateChange trap signifies that there has been a change in the state of an OSPF virtual 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")."

::= 3

ospfIfConfigError TRAP-TYPE
ENTERPRISE ospf
VARIABLES {
  ospfRouterId, -- The originator of the trap
  ospfIfIpAddress,
  ospfAddressLessIf,
  IpAddress, -- The source IP address
  ospfConfigErrorType, -- Type of error
An ospfIfConfigError trap signifies that a packet has been received on a non-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 adjacency from forming.

::= 4

ospfVirtIfConfigError TRAP-TYPE
ENTERPRISE ospf
VARIABLES {
  ospfRouterId, -- The originator of the trap
  ospfVirtAreaId,
  ospfVirtIfNeighbor,
  ospfConfigErrorType, -- Type of error
  ospfPacketType }
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 adjacency from forming."
::= 5

ospfIfAuthFailure TRAP-TYPE
ENTERPRISE ospf
VARIABLES {
  ospfRouterId, -- The originator of the trap
  ospfIfIpAddress,
  ospfAddressLessIf,
  IpAddress, -- The source IP address
  ospfConfigErrorType, -- authTypeMismatch or authFailure
  ospfPacketType }
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."
::= 6

ospfVirtIfAuthFailure TRAP-TYPE
ENTERPRISE ospf
VARIABLES {
  ospfRouterId, -- The originator of the trap
  ospfVirtAreaId,
  ospfVirtIfNeighbor,
  ospfConfigErrorType, -- authTypeMismatch or authFailure
  ospfPacketType }
ospfPacketType }
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."
::= 7

ospfIfRxBadPacket  TRAP-TYPE
ENTERPRISE      ospf
VARIABLES {
    ospfRouterId, -- The originator of the trap
    ospfIfIpAddress,
    ospfAddressLessIf,
    IpAddress,  -- The source IP address
    ospfPacketType }
DESCRIPTION
"An ospfIfRxBadPacket trap signifies that an OSPF packet has been received on a non-virtual interface that cannot be parsed."
::= 8

ospfVirtIfRxBadPacket  TRAP-TYPE
ENTERPRISE      ospf
VARIABLES {
    ospfRouterId, -- The originator of the trap
    ospfVirtAreaId,
    ospfVirtIfNeighbor,
    ospfPacketType }
DESCRIPTION
"An ospfVirtIfRxBadPacket trap signifies that an OSPF packet has been received on a virtual interface that cannot be parsed."
::= 9

ospfTxRetransmit  TRAP-TYPE
ENTERPRISE      ospf
VARIABLES {
    ospfRouterId, -- The originator of the trap
    ospfIfIpAddress,
    ospfAddressLessIf,
    ospfNbrRtrId, -- Destination
    ospfPacketType,
    ospfLsdbType,
    ospfLsdbLsid,
    ospfLsdbRouterId }
DESCRIPTION
"An ospfTxRetransmit trap signifies that an OSPF packet has been retransmitted on a non-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.

::= 10

ospfVirtIfTxRetransmit TRAP-TYPE
ENTERPRISE ospf
VARIABLES {
    ospfRouterId, -- The originator of the trap
    ospfVirtAreaId,
    ospfVirtIfNeighbor,
    ospfPacketType,
    ospfLsdbType,
    ospfLsdbLsid,
    ospfLsdbRouterId
}
DESCRIPTION
"An ospfTxRetransmit trap signifies than 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."

::= 11

ospfOriginateLsa TRAP-TYPE
ENTERPRISE ospf
VARIABLES {
    ospfRouterId, -- The originator of the trap
    ospfLsdbAreaId,  -- 0.0.0.0 for AS Externals
    ospfLsdbType,
    ospfLsdbLsid,
    ospfLsdbRouterId
}
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. Additionally, this trap does not include LSAs that are being flushed because they have reached MaxAge."

::= 12

ospfMaxAgeLsa TRAP-TYPE
ENTERPRISE ospf
VARIABLES {
    ospfRouterId, -- The originator of the trap
    ospfLsdbAreaId,  -- 0.0.0.0 for AS Externals
    ospfLsdbType,
    ospfLsdbLsid,
    ospfLsdbRouterId
}
DESCRIPTION
"An ospfMaxAgeLsa trap signifies that one of the LSA in the router’s link-state database has aged to MaxAge."
::= 13

ospfLsdbOverflow       TRAP-TYPE
ENTERPRISE      ospf
VARIABLES {
  ospfRouterId, -- The originator of the trap
  ospfExtLsdbLimit }
DESCRIPTION
"An ospfLsdbOverflow trap signifies that the number of
LSAs in the router's link-state database has exceeded
ospfExtLsdbLimit."
::= 14

ospfLsdbApproachingOverflow     TRAP-TYPE
ENTERPRISE      ospf
VARIABLES {
  ospfRouterId, -- The originator of the trap
  ospfExtLsdbLimit }
DESCRIPTION
"An ospfLsdbApproachingOverflow trap signifies that the
number of LSAs in the router's link-state database has exceeded
ninety percent of ospfExtLsdbLimit."
::= 15

END

7.0 Acknowledgements

This document was produced by the OSPF Working Group, chaired by
John Moy.

In addition, the comments of the following individuals are also
acknowledged:
   Dino Farinacci  cisco
   Vince Fuller    BARRNet
   Stan Froyd      ACC
   Dean Morris     Digital Equipment Corp.
   John Moy        Proteon, Inc.

8.0 References


   Information Base Internet Draft, Internet Engineering

[3] Information processing systems - Open Systems
   Interconnection - Specification of Abstract Syntax


