Network Working Group   A. Lindem (Editor)
Internet-Draft   Ericsson
Intended status: Standards Track   S. Mirtorabi
Expires: June 18, 2010   A. Roy
                                                            M. Barnes
                                                        Cisco Systems
                                                          R. Aggarwal
                                                      Juniper Networks
                                                    December 15, 2009

Support of address families in OSPFv3
draft-ietf-ospf-af-alt-10.txt

Abstract

   This document describes a mechanism for supporting multiple address
   families in OSPFv3 using multiple instances. It maps an address
   family (AF) to an OSPFv3 instance using the Instance ID field in the
   OSPFv3 packet header. This approach is fairly simple and minimizes
   extensions to OSPFv3 for supporting multiple AFs.

Status of this Memo

   This Internet-Draft is submitted to IETF in full conformance with the
   provisions of BCP 78 and BCP 79.

   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 will expire on June 18, 2010.

Copyright Notice

Table of Contents

1. Introduction

   OSPFv3 [OSPFV3] has been defined to support the base IPv6 unicast
   Address Family (AF). There are requirements to advertise other AFs
   in OSPFv3 including multicast IPv6, unicast IPv4, and multicast IPv4.
   This document supports these other AFs in OSPFv3 by mapping each AF
   to a separate Instance ID and OSPFv3 instance.

1.1. Design Considerations

   This section describes the rationale for using the multiple instance
   ID approach to support multiple address families in OSPFv3. As
   described earlier, OSPFv3 is designed to support multiple instances.
   Hence mapping an instance to an address family doesn't introduce any
   new mechanisms to the protocol. It minimizes the protocol extensions
   required and it simplifies the implementation. The presence of a
   separate link state database per address family is also easier to
   debug and operate. Additionally, it doesn't change the existing
   instance, area, and interface based configuration model in most
   OSPFv3 implementations.

1.2. Requirements notation

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

2. Protocol Details

   Currently the entire Instance ID number space is used for IPv6
   unicast. This specification assigns different Instance ID ranges to
   different AFs in order to support other AFs in OSPFv3. Each Instance
   ID implies a separate OSPFv3 instance with its own neighbor
   adjacencies, link state database, protocol data structures, and
   shortest path first (SPF) computation.

   Additionally, the current Link State Advertisements (LSAs) defined to
   advertise IPv6 unicast prefixes can be used to advertise prefixes
   from other AFs without modification.

   It should be noted that OSPFv3 runs on top of IPv6 and uses IPv6 link
   local addresses for OSPFv3 control packets. Therefore, it is
   required that IPv6 be enabled on an OSPFv3 link, although the link
   may not be participating in any IPv6 AFs.

2.1. Instance ID values for new AFs

   Instance ID zero is already defined by default for the IPv6 unicast
   AF. When this specification is used to support multiple AFs, we
   define the following ranges for different AFs. The first value of
   each range is the default value for the corresponding AF.

```
      Instance ID # 0    -  # 31     IPv6 unicast AF
      Instance ID # 32   -  # 63     IPv6 multicast AF
      Instance ID # 64   -  # 95     IPv4 unicast AF
      Instance ID # 96   -  # 127    IPv4 multicast AF
      Instance ID # 128  -  # 255    Unassigned
```

                       OSPFv3 Instance IDs

2.2. OSPFv3 Options Changes

   A new AF-bit is added to the OSPFv3 options field. The V6-bit is
   only applicable to the IPv6 unicast AF.

```
                         1                   2
        0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8  9  0  1  2  3
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+--+-+-+--+-+-+--+--+--+
       | | | | | | | | | | | | | | | |  |AF|*|*|DC|R|N|x | E|V6|
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+--+-+-+--+-+-+--+--+--+
```

                       The Options field

                       OSPFv3 Options

   V6-bit
      The V6 bit is used in OSPFv3 to exclude a node from IPv6 unicast
      route calculation but allow it in the SPF calculation for other
      address families. Since Instance ID now denotes the AF
      explicitly, this bit is ignored in AFs other than IPv6 unicast.

   AF-bit
      When an OSPFv3 router is supporting AFs as described in this
      specification, it MUST set the AF-bit in the OSPFv3 Options field
      of Hello Packets, Database Description packets, and LSAs.

2.3. Advertising Prefixes in AFs other than IPv6

   Each Prefix defined in OSPFv3 has a prefix length field. This
   facilitates advertising prefixes of different lengths in different
   AFs. The existing LSAs defined in OSPFv3 are used for this purpose
   and there is no need to define new LSAs.

   Prefixes which don't conform to OSPFv3 instance AF MUST be not be
   used in the route computation for the instance.

2.4. Changes to the Hello processing

   When an OSPFv3 router does not support this specification and it is
   configured with the corresponding Instance ID, packets could be black
   holed. This could happen due to misconfiguration or a router
   software downgrade. Black holing is possible because the router
   which doesn't support the AF can still be included in the SPF
   calculated path as long as it establishes adjacencies using the
   Instance ID corresponding to the AF. Note that Router-LSAs and
   Network-LSAs are AF independent.

   In order to avoid the above situation, hello processing is changed in
   order to only establish adjacencies with routers that have the AF-bit
   set in their Options field.

   Receiving Hello Packets is specified in section 3.2.2.1 of [OSPFV3].
   The following check is added to Hello reception:

   o When an OSPFv3 router participates in an AF (sets the AF-bit in
      Options field), it MUST discard Hello packets having the AF-bit
      clear in the Options field. The only exception is the Base IPv6
      unicast AF, where this check MUST NOT be done (for backward
      compatibility).

2.5. Next-Hop Calculation for IPv4 unicast and multicast AFs

   OSPFv3 runs on top of IPv6 and uses IPv6 link local addresses for
   OSPFv3 control packets and next-hop calculations. Although IPV6 link
   local addresses could be used as next-hops for IPv4 address families,
   it is desirable to have IPv4 next-hop addresses. For example, in the
   IPv4 multicast AF, the Protocol Independent Multicast (PIM) [PIM]
   neighbor address and the next-hop address should both be IPv4
   addresses in order for the Reverse Path Forwarding (RPF) lookup to
   work correctly. Troubleshooting is also easier when the prefix
   address and next-hop address are in the same AF.

   In order to achieve this, the link's IPv4 address will be advertised
   in the "link local address" field of the IPv4 instance's Link-LSA.
   This address is placed in the first 32 bit of "link local address"
   field and is used for IPv4 next-hop calculations. The remaining bits
   MUST be set to zero.

   We call the direct interface address (DIA) the address that is
   reachable directly via the link provided that a layer 3 to layer 2
   mapping is available. Note that there is no explicit need for the
   IPv4 link addresses to be on the same subnet. An implementation
   SHOULD resolve layer 3 to layer 2 mappings via Address Resolution
   Protocol (ARP) [ARP] or Neighbor Discovery (ND) [ND] for a DIA even
   if the IPv4 address is not on the same subnet as the router's
   interface IP address.

2.6. AS External LSA Forwarding Address for IPv4 Unicast and IPv4 Multicast AFs

   For OSPFv3, this address is an IPv6 host address (128 bits). If
   included, data traffic for the advertised destination will be
   forwarded to this address. For IPv4 unicast and IPv4 multicast AFs,
   the Forwarding Address in AS-external-LSAs MUST encode an IPv4
   address. To achieve this, the IPv4 Forwarding Address is advertised
   by placing it in the first 32 bits of the Forwarding Address field in
   the AS-external-LSAs. The remaining bits MUST be set to zero.

2.7. Database Description Maximum Transmission Unit (MTU) Specification for Non-IPv6 AFs

   For address families other than IPv6, both the MTU for the instance
   address family and IPv6 MTU used for OSPFv3 maximum packet
   determination MUST be considered. The MTU in the Database
   Description packet MUST always contain the MTU corresponding to the
   advertised address family. For example, if the instance corresponds
   to an IPv4 address family, the IPv4 MTU for the interface MUST be
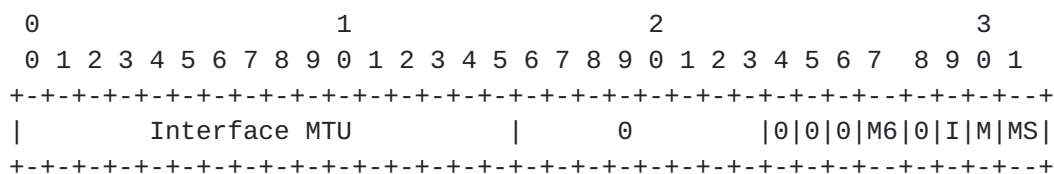   specified in the interface MTU field. As specified section 10.6 of

[OSPFV2], the Database Description packet will be rejected if the MTU
is greater than the receiving interface's MTU for the address family
corresponding to the instance. This behavior will assure an
adjacency is not formed and address family specific routes are not
installed over a path with conflicting MTUs.

The value used for OSPFv3 maximum packet size determination MUST also
be compatible for an adjacency to be established. Since only a
single MTU field is specified, the M6-bit is defined by this
specification. If the M6-bit is clear, the specified MTU SHOULD also
be checked against the IPv6 MTU and the Database Description packet
SHOULD be rejected if the MTU is larger than the receiving
interface's IPv6 MTU. An OSPFv3 router SHOULD NOT set the M6-bit if
its IPv6 MTU and address family specific MTU are the same.

If the IPv6 and IPv4 MTUs differ, the M6-bit MUST be set for non-IPv6
address families. If the M6-bit is set, the IPv6 MTU is decided by
the presence or absence of an IPv6 MTU TLV in the LLS [LLS] block.
If this TLV is present, it carries the IPv6 MTU that SHOULD be
compared with local IPv6 MTU. If this TLV is absent, the minimum
IPv6 MTU of 1280 octets SHOULD be used for the comparison (refer to
[IPV6]).

If the M6-bit is set in a received Database Description packet for a
non-IPv6 address family, the receiving router MUST NOT check the
Interface MTU in the Database Description packet against the
receiving interface's IPv6 MTU.

The figure below graphically depicts the changed fields in the octets
20-23 of the OSPFv3 Database Description Packet:

```
  0                   1                   2                   3
  0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |        Interface MTU          |       0       |0|0|0|M6|0|I|M|MS|
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

              OSPFv3 Database Description Packet Changes

The changed fields in the Database Description packet are described
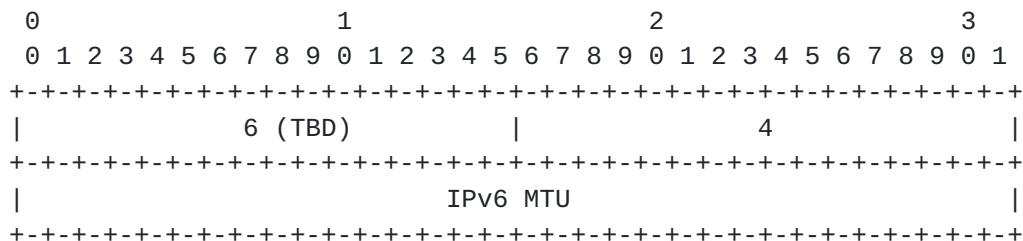below. The remaining fields are unchanged from [OSPFV3].

Interface MTU
   The size in octets of the largest AF specific datagram that can be
   sent on the associated interface without fragmentation. The MTUs
   of common Internet link types can be found in Table 7-1 of
   [MTUDISC]. Interface MTU SHOULD be set to 0 in Database

      Description packets sent over virtual links.

   M6-bit
      The IPv6 MTU bit - this bit indicates the sender is using a
      different IPv6 MTU than the MTU for the AF.

   An IPv6 MTU TLV can be optionally carried in LLS block as described
   above. This TLV carries the IPv6 MTU for the interface. The length
   field of the TLV is set to 4 bytes.


```
     0                   1                   2                   3
     0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |              6 (TBD)           |               4               |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |                            IPv6 MTU                            |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```


                        Format of IPv6 MTU TLV

   Only one instance of the IPv6 MTU TLV MAY appear in the LLS block.
   Instances subsequent to the first are not processed and the LLS
   inconsistency SHOULD be logged.

2.8. Operation over Virtual Links

   OSPFv3 control packets sent over a virtual link are IPv6 packets and
   may traverse multiples hops. Therefore, there MUST be a global IPv6
   address associated with the virtual link so that OSPFv3 control
   packets are forwarded correctly by the intermediate hops between
   virtual link endpoints. Although this requirement can be satisfied
   in IPv6 unicast AFs, it will not function in other AFs as there will
   not be a routable global IPv6 address or forwarding path. Therefore,
   virtual links are not supported in AFs other than IPv6 Unicast.

[3](#). **Backward Compatibility**

   All modifications to OSPFv3 apply exclusively to the support address
   families other than IPv6 unicast using multiple OSPFv3 instances as
   described in this specification. They are not applicable to IPv6
   unicast topologies and do not preclude future single instance
   mechanisms for supporting multiple address families.

   In this section, we will define a non-capable OSPFv3 router as one
   not supporting this specification. When multiple AFs are supported
   as defined herein, each new AF will have a corresponding Instance ID
   and can interoperate with the existing non-capable OSPFv3 routers in
   an IPv6 unicast topology. Furthermore, when a non-capable OSPFv3
   router uses an Instance ID that is reserved for a given AF, no
   adjacency will be formed with this router since the AF-bit in the
   Options field will be clear in its OSPFv3 Hello packets. Therefore,
   there are no backward compatibility issues. AFs can be gradually
   deployed without disturbing OSPFv3 routing domains with non-capable
   OSPFv3 routers.

4. Security Considerations

   IPsec [IPsec] can be used for OSPFv3 authentication and
   confidentiality as described in [OSPFV3-AUTH]. When multiple OSPFv3
   instances use the same interface, they all MUST use the same Security
   Association (SA), since the SA selectors do not provide selection
   based on OSPFv3 header fields, such as the Instance ID. This
   restriction is documented in section 8 of [OSPFV3-AUTH].

   Security considerations for the OSPFv3 are covered in [OSPFV3].

5. IANA Considerations

The following IANA assignments are to be made from existing
registries:

The AF-bit is assigned from OSPFv3 Options field as defined in
Section 2.2.

IANA is requested to create a new registry, "OSPFv3 Instance ID
Address Family Values", for assignment of the mapping of OSPFv3
Instance ID to address families when this specification is used to
support multiple address families. Note that the Instance ID field
MAY be used for applications other than the support of multiple
address families. However, if it is being used for address families
as described in this specification, the assignments herein SHOULD be
honored.

Value/Range	Designation	Assignment Policy
0	Base IPv6 Unicast AF	Already assigned
1-31	IPv6 Unicast AFs dependent on local policy	Already assigned
32	Base IPv6 Multicast	Already assigned
33-63	IPv6 Multicast AFs dependent on local policy	Already assigned
64	Base IPv4 Unicast AF	Already assigned
65-95	IPv4 Unicast AFs dependent on local policy	Already assigned
96	Base IPv4 Multicast	Already assigned
97-127	IPv4 Multicast AFs dependent on local policy	Already assigned
128-255	Unassigned	Standards Action

OSPFv3 Address Family Use of Instance IDs

   o Instance IDs 0-127 are assigned by this specification.

   o Instance IDs in the range 128-255 are not assigned at this time.
      Before any assignments can be made in this range, there MUST be a
      Standards Track RFC including IANA Considerations explicitly
      specifying the AF Instance IDs being assigned.

   M6-Bit is assigned as defined in Section 2.7.

   A TLV type for IPv6 MTU TLV is assigned from OSPF LLS TLVs registry.

6. References

6.1. Normative References

[IPV6]   Deering, S. and R. Hinden, "Internet Protocol, Version 6
            (IPv6) Specification", RFC 2460, December 1998.

[IPsec]   Kent, S. and K. Seo, "Security Architecture for the
            Internet Protocol", RFC 4301, December 2005.

[OSPFV2]   Moy, J., "OSPF Version 2", RFC 2328, April 1998.

[OSPFV3]   Coltun, R., Ferguson, D., Moy, J., and A. Lindem, "OSPF
            for IPv6", RFC 5340, July 2008.

[OSPFV3-AUTH]
            Gupta, M. and S. Melam, "Authentication/Confidentiality
            for OSPFv3", RFC 4552, June 2006.

[RFC-KEYWORDS]
            Bradner, S., "Key words for use in RFC's to Indicate
            Requirement Levels", RFC 2119, March 1997.

6.2. Informative References

[ARP]   Plummer, D., "An Ethernet Address Resolution Protocol",
            RFC 826, November 1982.

[LLS]   Zinin, A., Roy, A., Nguyen, L., Friedman, B., and D.
            Young, "OSPF Link-local Signaling", RFC 5613, August 2009.

[MTUDISC]   Mogul, J. and S. Deering, "Path MTU Discovery", RFC 1191,
            November 1990.

[ND]   Narten, T., Nordmark, E., Simpson, W., and H. Soliman,
            "Neighbor Discovery for IP Version 6 (IPv6)", RFC 4861,
            September 2007.

[PIM]   Fenner, B., Handley, M., Holbrook, H., and I. Kouvelas,
            "Protocol Independent Multicast - Sparse Mode (PIM-SM):
            Protocol Specification (Revised)", RFC 4601, August 2006.

Appendix A. Acknowledgments

    The RFC text was produced using Marshall Rose's xml2rfc tool.

    Thanks to Tom Henderson and the folks at Boeing for implementing in
    the Quagga routing suite, http:www.quagga.net.

    Thanks to Nischal Sheth for review and comments.

    Thanks to Christian Vogt for comments during the Gen-ART review.

    Thanks to Adrian Farrel for comments during the IESG review.

Authors' Addresses

    Acee Lindem
    Ericsson
    102 Carric Bend Court
    Cary, NC 27519
    USA

    Email: acee.lindem@ericsson.com

    Sina Mirtorabi
    Cisco Systems
    3 West Plumeria Drive
    San Jose, CA 95134
    USA

    Email: smirtora@cisco.com

    Abhay Roy
    Cisco Systems
    225 West Tasman Drive
    San Jose, CA 95134
    USA

    Email: akr@cisco.com

    Michael Barnes
    Cisco Systems
    225 West Tasman Drive
    San Jose, CA 95134
    USA

    Email: mjbarnes@cisco.com

    Rahul Aggarwal
    Juniper Networks
    1194 N. Mathilda Ave.
    Sunnyvale, CA 94089
    USA

    Email: rahul@juniper.net