BGP Neighbor Autodiscovery
draft-xu-idr-neighbor-autodiscovery-06

Abstract

   BGP has been used as the underlay routing protocol in many hyper-
   scale data centers. This document proposes a BGP neighbor
   autodiscovery mechanism that greatly simplifies BGP deployments.
   This mechanism is very useful for those hyper-scale data centers
   where BGP is used as the underlay routing protocol.

Requirements Language

   The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT",
   "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this
   document are to be interpreted as described in RFC 2119 [RFC2119].

Status of This Memo

Table of Contents

1. Introduction

   BGP has been used as the underlay routing protocol instead of IGP in
   many hyper-scale data centers [RFC7938]. Furthermore, there is an
   ongoing effort to leverage BGP link-state distribution mechanism to
   achieve BGP-SPF [I-D.keyupate-lsvr-bgp-spf]. However, BGP is not
   good as an IGP from the perspective of deployment automation and
   simplicity. For instance, the IP address and the Autonomous System
   Number (ASN) of each and every BGP neighbor have to be manually
   configured on BGP routers although these BGP peers are directly
   connected. Furthermore, for those BGP routers with multiple physical
   links being connected, it's usually not ideal to establish BGP
   sessions over their directly connected interface addresses because
   the BGP update volume would be unnecessarily increased, meanwhile, it
   may not be suitable to configure those links as a Link Aggregation
   Group (LAG) due to some reasons. As a result, it's more common that

loopback interface addresses of those directly connected BGP peers
are used for BGP session establishment purpose. To make those
loopback addresses of directly connected BGP peers reachable from one
another, either static routes have to be configured or some kind of
IGP has to be enabled. The former is not good from the network
automation perspective while the latter is not good from the network
simplification perspective (i.e., running less routing protocols).

This draft specifies a BGP neighbor autodiscovery mechanism by
borrowing some ideas from the Label Distribution Protocol (LDP)
[RFC5036] . More specifically, directly connected BGP routers could
automatically discovery each other through the exchange of the to-be-
defined BGP Hello messages. The BGP session establishment process as
defined in [RFC4271] could be triggered once directly connected BGP
neighbors are discovered from one another. Note that the BGP session
should be established over the discovered the peering address of the
BGP neighbor and in most cases the peering address is a loopback
address. In addition, to eliminate the need of configuring static
routes or enabling IGP for the loopback addresses, a certain type of
routes towards the BGP neighbor's loopback addresses as advertised as
peering addresses are dynamically instantiated once the BGP neighbor
has been discovered. The administrative distance of such type of
routes MUST be smaller than their equivalents that are learnt by the
regular BGP update messages . Otherwise, circular dependency would
occur once these loopback addresses are advertised via the regular
BGP updates.

2. Terminology

This memo makes use of the terms defined in [RFC4271].

3. BGP Hello Message Format

To automatically discover directly connected BGP neighbors, a BGP
router periodically sends BGP HELLO messages out those interfaces on
which BGP neighbor autodiscovery are enabled. The BGP HELLO message
MUST sent as a UDP packet with a destination port of TBD (179 is the
preferred port number value) addressed for the "all routers on this
subnet" group multicast address (i.e., 224.0.0.2 in the IPv4 case and
FF02::2 in the IPv6 case). The IP source address is set to the
address of the interface over which the message is sent out.

The HELLO message contains the following fields:
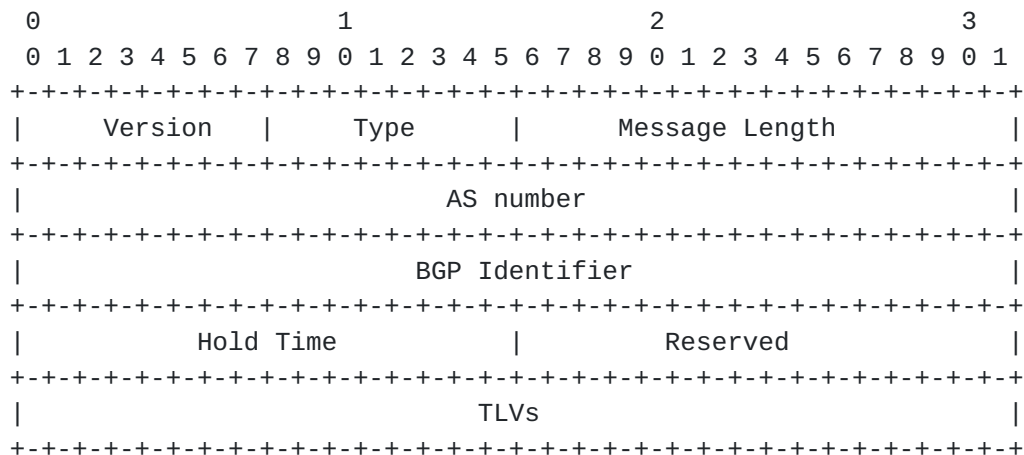
```
       0                   1                   2                   3
       0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
      +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
      |     Version   |      Type     |        Message Length         |
      +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
      |                          AS number                           |
      +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
      |                        BGP Identifier                        |
      +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
      |          Hold Time          |            Reserved            |
      +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
      |                             TLVs                             |
      +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
                       Figure 1: BGP Hello Message
```

   Version: This 1-octet unsigned integer indicates the protocol
   version number of the message. The current BGP version number is
   4.

   Type: The type of BGP message (Hello - TBD value from BGP Message
   Types Registry)

   Message Length: This 2-octet unsigned integer specifies the length
   in octets of the TLVs field.

   AS number: AS Number of the Hello message sender.

   BGP Identifier: BGP Identifier of the Hello message sender.

   Hold Time: Hello hold timer in seconds. Hello Hold Time specifies
   the time the receiving BGP peer will maintain its record of Hellos
   from the sending BGP peer without receipt of another Hello. The
   RECOMMENDED default value is 15 seconds. A value of 0 means that
   the receiving BGP peer should maintain its record until the link
   is UP.

   Reserved: SHOULD be set to 0 by sender and MUST be ignored by
   receiver.

   TLVs: This field contains one or more TLVs as described below.

The Accepted ASN List TLV is an optional TLV that is used to signal
the AS numbers from which the router would accept BGP sessions. When
not signaled, it indicates that the router will accept BGP peering
from any ASN from its neighbors. Only a single instance of this TLV
is included and its format is shown below.

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|            Type               |            Length             |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                 Accepted ASN List(variable)                   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
            Figure 2: Accepted ASN List TLV
```
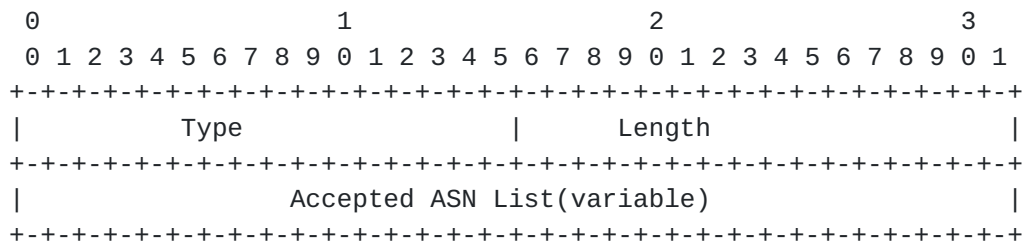
   Type: TBD1

   Length:Specifies the length of the Value field in octets.

   Accepted ASN-List: This variable-length field contains one or more
   accepted 4-octet ASNs.

The Peering Address TLV is used to indicate to the neighbor the
address to which they should establish BGP session. For each peering
address, the router can specify its supported AFI/SAFI(s). When the
AFI/SAFI values are specified as 0/0, then it indicates that the
neighbor can attempt for negotiation of any AFI/SAFIs. The
indication of AFI/SAFI(s) in the Peering Address TLV is not intended
as an alternative for the MP capabilities negotiation mechanism.

The Peering Address TLV format is shown below and at least one
instance of this TLV MUST be present.

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|           Type                |            Length             |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  Flags        | No. AFI/SAFI  |      Reserved                 |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|      Address (4-octet or 16-octet)                            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+

+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|            AFI                |    SAFI       | ...
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+

+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|      sub-TLVs ...
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
            Figure 3: Peering Address TLV
```
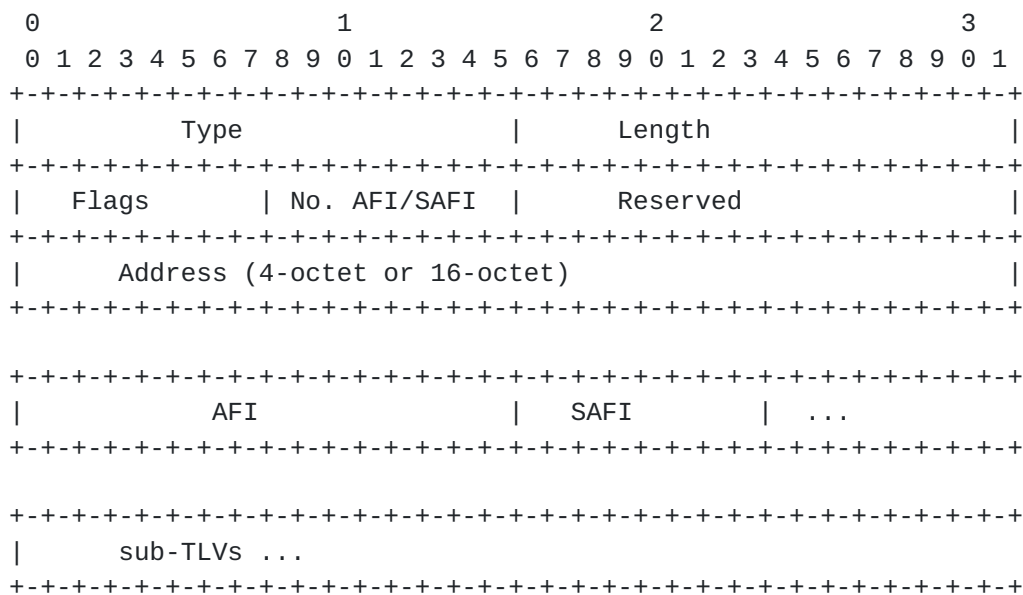
   Type: TBD2

   Length:Specifies the length of the Value field in octets.

   Flags : Current defined bits are as follows. All other bits
   SHOULD be cleared by sender and MUST be ignored by receiver.

      Bit 0x1 - address is IPv6 when set and IPv4 when clear

   Number of AFI/SAFI: indicates the number of AFI/SAFI pairs that
   the router supports on the given peering address.

   Reserved: sender SHOULD set to 0 and receiver MUST ignore.

   Address: This 4 or 16 octect field indicates the IPv4 or IPv6
   address which is used for establishing BGP sessions.

   AFI/SAFI : one or more pairs of these values that indicate the
   supported capabilities on the peering address.

   Sub-TLVs : currently none defined

 When the Peering Address used is not the directly connected interface
 address (e.g. when it is a loopback address) then local prefix(es)
 that cover the peering address(es) MUST be signaled by the router.
 This allows the neighbor to learn these local prefix(es) and to
 program routes for them over the directly connected interfaces over
 which they are being signalled. The Local Prefixes TLV is used to
 only signal prefixes that are locally configured on the router and
 its format is as shown below.

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|           Type               |           Length               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|      No. of IPv4 Prefixes    |       No. of IPv6 Prefixes     |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+

+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                       IPv4 Prefix                             |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  Prefix Mask  | ...
+-+-+-+-+-+-+-+-+-+

+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                       IPv6 Prefix                             |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  Prefix Mask  | ...
+-+-+-+-+-+-+-+-+-+

+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  sub-TLVs ...
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 4: Local Prefixes TLV

Type: TBD3

Length:Specifies the length of the Value field in octets

No. of IPv4 Prefixes : specifies the number of IPv4 prefixes.
When value is 0, then it indicates no IPv4 Prefixes are present.

No. of IPv6 Prefixes : specifies the number of IPv6 prefixes.
When value is 0, then it indicates no IPv6 Prefixes are present.

IPv4 Prefix Address & Prefix Mask: Zero or more pairs of IPv4
prefix address and their mask.

IPv6 Prefix Address & Prefix Mask: Zero or more pairs of IPv6
prefix address and their mask.

Sub-TLVs : currently none defined

The Link Attributes TLV is a mandatory TLV that signals to the
neighbor the link attributes of the interface on the local router. A
single instance of this TLV MUST be present in the message. The Link
Attributes TLV is as shown below.

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|            Type              |           Length              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|      Local Interface ID      |    Flags    |    Reserved     |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|    No. of IPv4 Addresses     |    No. of IPv6 Addresses      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+

+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                   IPv4 Local Address                         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Prefix Mask  | ...
+-+-+-+-+-+-+-+-+-+

+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                   IPv6 Local Address                         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Prefix Mask  | ...
+-+-+-+-+-+-+-+-+-+

+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| sub-TLVs ...
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

                  Figure 5: Link Attributes TLV

Type: TBD4

Length:Specifies the length of the Value field in octets

Local Interface ID : the local interface ID of the interface (e.g.
the MIB-2 ifIndex)

Flags : Currently defined bits are as follows. Other bits SHOULD
be cleared by sender and MUST be ignored by receiver.

   Bit 0x1 - indicates link is enabled for IPv4

   Bit 0x2 - indicates link is enabled for IPv6

Reserved: SHOULD be set to 0 by sender and MUST be ignored by
receiver.

No. of IPv4 Addresses : specifies the number of IPv4 local
addresses on the interface. When value is 0, then it indicates no
IPv4 Prefixes are present or the interface is IP unnumbered.

   No. of IPv6 Addresses : specifies the number of IPv6 Global
   addresses on the interface. When value is 0, then it indicates no
   IPv6 Global Prefixes are present or the interface is only
   configured with IPv6 link-local addresses

   IPv4 Address & Mask: Zero or more pairs of IPv4 address and their
   mask.

   IPv6 Address & Mask: Zero or more pairs of IPv6 address and their
   mask.

   Sub-TLVs : currently none defined

The Neighbor TLV is used by a BGP router to indicate the peering
address and information about the neighbors that have been discovered
by the router on the specific link and their status. The BGP session
establishment process begins when both the neighbors accept each
other over at least one underlying inter-connecting link between
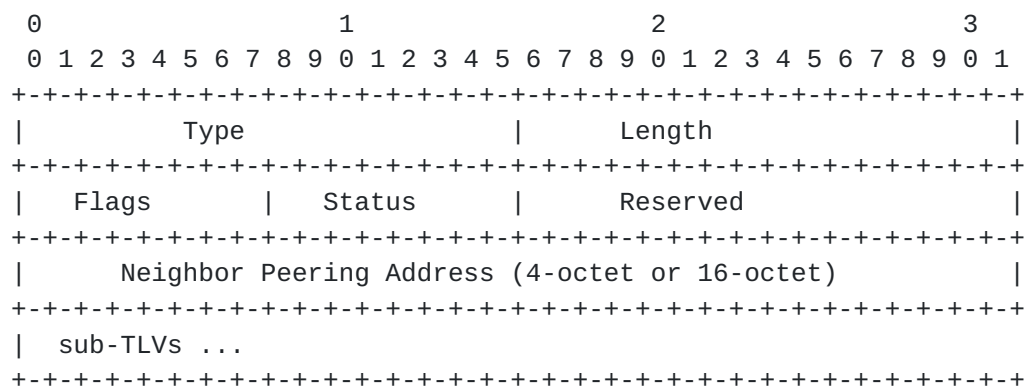them. The Neighbor TLV format is as shown below.

```
    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
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |           Type             |            Length                |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |   Flags       |   Status    |        Reserved                 |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |       Neighbor Peering Address (4-octet or 16-octet)         |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |   sub-TLVs ...
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

                   Figure 6: Neighbor TLV

   Type: TBD5

   Length:Specifies the length of the Value field in octets

   Flags : Currently defined 0x1 bit is clear when Peering Address is
   IPv4 and set when IPv6. Other bits SHOULD be clear by sender and
   MUST be ignored by receiver.

   Status : Indicates the status code of the peering for the
   particular session over this link. The following codes are
   currently defined

> 0 - Indicates 1-way detection of the peer
>
> 1 - Indicates rejection of the peer due to local policy reasons
> (i.e. local router would not be initiating or accepting session
> to this neighbor)
>
> 2 - Indicates 2-way detection of the peering by both neighbors
>
> 3 - Indicates that the BGP peering session has been established
> between the neighbors and that this link would be utilized for
> forwarding to the peer BGP nexthop

Reserved: SHOULD be set to 0 by sender and MUST be ignored by
receiver.

Neighbor Peering Address: This 4 or 16 octect field indicates the
IPv4 or IPv6 peering address of the neighbor for which peering
status is being reported.

Sub-TLVs : currently none defined

[4](#). Hello Message Procedure

A BGP peer receiving Hellos from another peer maintains a Hello
adjacency corresponding to the Hellos. The peer maintains a hold
timer with the Hello adjacency, which it restarts whenever it
receives a Hello that matches the Hello adjacency. If the hold timer
for a Hello adjacency expires the peer discards the Hello adjacency.

We recommend that the interval between Hello transmissions be at most
one third of the Hello hold time.

A BGP session with a peer has one or more Hello adjacencies.

A BGP session has multiple Hello adjacencies when a pair of BGP peers
is connected by multiple links that have the same connection address
(e.g., multiple point-to-point links between a pair of routers). In
this situation, the Hellos a BGP peer sends on each such link carry
the same Peering Address. In addition, to eliminate the need of
configuring static routes or enabling IGP for advertising the
loopback addresses, a certain type of routes towards the BGP
neighbor's loopback addresses (i.e. carried in the Local Prefixes
TLV) could be dynamically created once the BGP neighbor has been
discovered. The administrative distance of such type of routes MUST
be smaller than their equivalents which are learnt via the normal BGP
update messages. Otherwise, circular dependency problem would occur
once these loopback addresses are advertised via the normal BGP
update messages as well.

   BGP uses the regular receipt of BGP Hellos to indicate a peer's
   intent to keep BGP session identified by the Hello. A BGP peer
   maintains a hold timer with each Hello adjacency that it restarts
   when it receives a Hello that matches the adjacency. If the timer
   expires without receipt of a matching Hello from the peer, BGP
   concludes that the peer no longer wishes to keep BGP session for that
   link or that the peer has failed. The BGP peer then deletes the
   Hello adjacency. The route towards the BGP neighbor's loopback
   address that had been dynamically created due to that BGP Hello
   adjacency SHOULD be deleted accordingly. When the last Hello
   adjacency for an BGP session is deleted, the BGP peer terminates the
   BGP session and closing the transport connection.

5. Contributors

   Satya Mohanty
   Cisco
   Email: satyamoh@cisco.com

   Shunwan Zhuang
   Huawei
   Email: zhuangshunwan@huawei.com

6. Acknowledgements

   The authors would like to thank Enke Chen for his valuable comments
   and suggestions on this document.

7. IANA Considerations

7.1. BGP Hello Message

   This document requests IANA to allocate a new UDP port (179 is the
   preferred number) and a BGP message type code for BGP Hello message.

    Value   TLV Name   Reference
    -----   -----------------------------------   -------------
    Service Name: BGP-HELLO
    Transport Protocol(s): UDP
    Assignee: IESG <iesg@ietf.org>
    Contact: IETF Chair <chair@ietf.org>.
    Description: BGP Hello Message.
    Reference: This document -- draft-xu-idr-neighbor-autodiscovery.
    Port Number: TBD1 (179 is the preferred value) -- To be assigned by IANA.

7.2. TLVs of BGP Hello Message

   This document requests IANA to create a new registry "TLVs of BGP
   Hello Message" with the following registration procedure:

             Registry Name: TLVs of BGP Hello Message.

   Value   TLV Name   Reference
   -------   -- -------------
         0   Reserved   This document
         1   Accepted ASN List   This document
         2   Peering Address   This document
         3   Local Prefixes   This document
         4   Link Attributes   This document
         5   Neighbor   This document
    6-65500   Unassigned
 65501-65534   Experimental   This document
     65535   Reserved   This document

8. Security Considerations

   For security purposes, BGP speakers usually only accept TCP
   connection attempts to port 179 from the specified BGP peers or those
   within the configured address range. With the BGP neighbor auto-
   discovery mechanism, it's configurable to enable or disable sending/
   receiving BGP hello messages on the per-interface basis and BGP hello
   messages are only exchanged between physically connected peers that
   are trustworthy. Therefore, the BGP neighbor auto-discovery
   mechanism doesn't introduce additional security risks associated with
   BGP.

   In addition, for the BGP sessions with the automatically discovered
   peers via the BGP hello messages, the TTL of the TCP/BGP messages
   (dest port=179) MUST be set to 255. Any received TCP/BGP message
   with TTL being less than 254 MUST be dropped according to [RFC5082].

9. References

9.1. Normative References

   [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate
              Requirement Levels", BCP 14, RFC 2119,
              DOI 10.17487/RFC2119, March 1997,
              <https://www.rfc-editor.org/info/rfc2119>.

   [RFC4271] Rekhter, Y., Ed., Li, T., Ed., and S. Hares, Ed., "A
              Border Gateway Protocol 4 (BGP-4)", RFC 4271,
              DOI 10.17487/RFC4271, January 2006,
              <https://www.rfc-editor.org/info/rfc4271>.

   [RFC5036] Andersson, L., Ed., Minei, I., Ed., and B. Thomas, Ed.,
              "LDP Specification", RFC 5036, DOI 10.17487/RFC5036,
              October 2007, <https://www.rfc-editor.org/info/rfc5036>.

   [RFC5082] Gill, V., Heasley, J., Meyer, D., Savola, P., Ed., and C.
              Pignataro, "The Generalized TTL Security Mechanism
              (GTSM)", RFC 5082, DOI 10.17487/RFC5082, October 2007,
              <https://www.rfc-editor.org/info/rfc5082>.

   [RFC8279] Wijnands, IJ., Ed., Rosen, E., Ed., Dolganow, A.,
              Przygienda, T., and S. Aldrin, "Multicast Using Bit Index
              Explicit Replication (BIER)", RFC 8279,
              DOI 10.17487/RFC8279, November 2017,
              <https://www.rfc-editor.org/info/rfc8279>.

9.2. Informative References

   [I-D.keyupate-lsvr-bgp-spf]
              Patel, K., Lindem, A., Zandi, S., and W. Henderickx,
              "Shortest Path Routing Extensions for BGP Protocol",
              draft-keyupate-lsvr-bgp-spf-00 (work in progress), March
              2018.

   [RFC7938] Lapukhov, P., Premji, A., and J. Mitchell, Ed., "Use of
              BGP for Routing in Large-Scale Data Centers", RFC 7938,
              DOI 10.17487/RFC7938, August 2016,
              <https://www.rfc-editor.org/info/rfc7938>.

Authors' Addresses

   Xiaohu Xu
   Alibaba Inc

   Email: xiaohu.xxh@alibaba-inc.com

   Kunyang Bi
   Huawei

   Email: bikunyang@huawei.com

   Jeff Tantsura
   Nuage Networks

      Email: jefftant.ietf@gmail.com

   Nikos Triantafillis
   LinkedIn

      Email: nikos@linkedin.com

   Ketan Talaulikar
   Cisco

      Email: ketant@cisco.com