Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: January 17, 2019

X. Xu
Alibaba Inc
K. Talaulikar
Cisco Systems
K. Bi
Huawei
J. Tantsura
Nuage Networks
N. Triantafillis
July 16, 2018

BGP Neighbor Auto-Discovery
draft-xu-idr-neighbor-autodiscovery-09

Abstract

BGP is being used as the underlay routing protocol in some large-scaled data centers (DCs). Most popular design followed is to do hop-by-hop external BGP (eBGP) session configurations between neighboring routers on a per link basis. The provisioning of BGP neighbors in routers across such a DC brings its own operational complexity.

This document introduces a BGP neighbor discovery mechanism that greatly simplifies BGP operations in such DC and other networks by automatic setup of BGP sessions between neighbor routers using this mechanism.

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

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

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at https://datatracker.ietf.org/drafts/current/.

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."

This Internet-Draft will expire on January 17, 2019.

Table of Contents

1. Introduction

BGP is being used as the underlay routing protocol instead of link-state routing protocols like IS-IS and OSPF in some large-scale data centers (DCs). [RFC7938] describes the design, configuration and operational aspects of using BGP in such networks. The most popular design scheme involves the setup of external BGP (eBGP) sessions over individual links between directly connected routers using their interface addresses. Such BGP neighbor provisioning requires provisioning of the neighbor IP address and Autonomous System (AS) Number (ASN) for each and every BGP neighbor on every link address. As a DC fabric comprising of topology described in [RFC7938] grows with addition of new leafs, spines and links between them, the BGP provisioning needs to be carefully setup. Unlike with the link-state protocols, there is no automatic discovery of neighbors simply by adding links and nodes in the fabric and route exchange over them getting enabled seamlessly in the case of BGP.

In some DC designs with BGP, multiple links are added between a leaf and spine to add additional bandwidth. Use of link-aggregation at Layer 2 level may not be desirable in such cases due to the risk of flow polarization on account of a mix of ECMP at Layer 2 and Layer 3 levels. In such cases, one option is for a eBGP sessions to be setup between two BGP neighbors over each of the links between them. In such a case, the BGP session scale and the resultant increase in update processing may pose scalability challenges. A second option is for a single eBGP session to be setup between the loopback IP addresses between the neighbor and then configure some static routes for it pointing over the underlying links as ECMP. In this option there is an additional provisioning task introduced in the form of static routing.

Furthermore, there is also a need for BGP to be able to describe its links and its neighbors on its directly connected links and export this information via BGP-LS [RFC7752] to provide a detail link-level topology view using a standards based mechanism of a data center running only BGP. The ability of BGP in discovering its neighbors over its links, monitoring their liveliness and learning the link attributes (such as addresses) is required for the conveying the link-state topology in a BGP network. This information can be leveraged by the BGP-SPF proposal [I-D.ietf-lsvr-bgp-spf] which introduces link-state routing capabilities in BGP. This information can also be leveraged to convey the link-state topology in a network

running traditional BGP routing using BGP-LS as described in [I-D.ketant-idr-bgp-ls-bgp-only-fabric] and to enabled end to end traffic engineering use-cases spanning across DCs and the core/access networks.

2. Terminology

This memo makes use of the terms defined in [RFC4271] and [RFC7938] .

3. Overview

At a high level, this specification introduces the use of UDP based BGP Hello messages to be exchanged between directly connected BGP routers for neighbor discovery.

1. Information is exchanged between BGP routers on a per link basis leading to discovery of each others peering address and other information.

2. The TCP session establishment for the BGP protocol operation and the BGP routing exchange over these sessions can then follow without any change/modification from the existing BGP protocol operations as specified in [RFC4271].

3. As part of the neighbor information exchange the route to a neighbor's peering address is also automatically setup pointing over the links over which the neighbor is discovered.

4. This route is used for both the BGP TCP session establishment as well as for resolution of the BGP next-hop (NH) for the routes learnt via the neighbor instead of an underlying IGP or static route.

Auto-discovery of BGP neighbors and their liveness detection may be performed via different mechanisms. This document prefers the use of an extension to BGP protocol since the deployments and use-cases targeted (i.e. large-scale DCs) are already running BGP as their routing protocol. Extending BGP with neighbor discovery capabilities is operationally and implementation wise a simpler approach than requiring a new or an additional protocol to be first extended to do this functionality (to exchange BGP-specific parameters) and then also integrated its operations with BGP protocol operations.

Following are the key objectives and goals of the BGP neighbor discovery mechanism proposed in this document:

- Existing BGP update processing is unchanged

- Minimal changes for integration of the neighbor discovery state machine with the existing BGP Peer state machine for auto-discovered neighbors only

- Auto-discovery mechanism is restricted to directly connected BGP speakers only and uses link-local multicast addresses only for the hello messaging

- Liveness detection is used for monitoring the BGP adjacency status for directly connected BGP routers over individual links and is BGP specific. It is not intended to replace the functionality for existing generic mechanisms like BFD and LLDP.

- Hello processing is separate from the core BGP protocol operations such that BGP route processing scale and performance is not impacted

The BGP neighbor discovery mechanism defined in this document borrows ideas from the Label Distribution Protocol (LDP) [RFC5036]. However, most importantly, only the concept of link-local signaling based neighbor discovery is borrow while the discovery aspect for targeted LDP sessions does not apply to this BGP neighbor discovery mechanism.

The further sections in this document first describe the newly introduced message formats and TLVs and then go on to describe the procedures of the BGP neighbor discovery mechanism and its integration with the base BGP protocol mechanism as specified in [RFC4271].

The operational and management aspects of the BGP neighbor discovery mechanism are described in Section 10.

4. UDP Message Header

The BGP neighbor discovery mechanism will operate using UDP messages. The UDP port of TBD (179 is the preferred port number to be assigned as specified in Section 11) is used which is same as the TCP port 179 used by BGP. The BGP UDP message common header format is specified as follows:

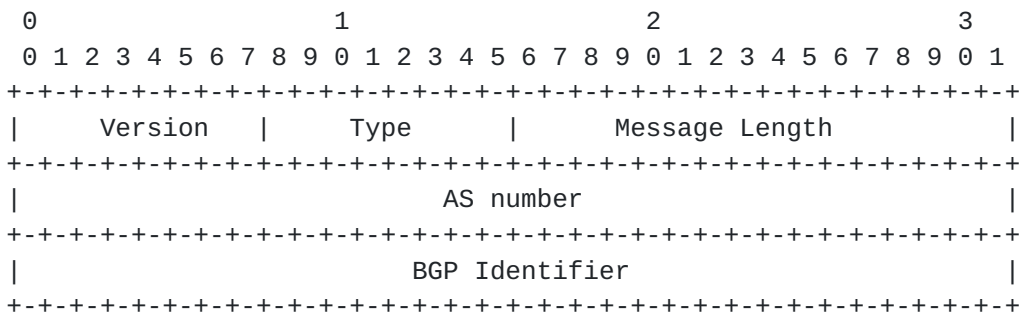

Figure 1: BGP UDP Message Header

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

Message Length: This 2-octet unsigned integer specifies the length in octets of the entire BGP UDP message including the header.

AS number: AS Number of the UDP message sender.

BGP Identifier: BGP Identifier of the UDP message sender.

BGP UDP messages can be sent using either IPv4 or IPv6 depending on the address used for session establishment and provisioned on the interfaces over which these messages are sent.

5. Hello Message Format

A BGP router uses UDP based Hello messages to automatically discover directly connected BGP neighbors and to check their liveliness. The Hello messages and the BGP neighbor discovery mechanism operates only on those interfaces where it is specifically enabled on. The BGP neighbor discovery mechanism is intend for link-local signaling between directly connected BGP nodes and hence the BGP Hello messages MUST be addressed to 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) and the TTL for the IP packets SHOULD be set to 1. The IP source address MUST be set to the address of the interface over which the message is sent out which would be the primary interface address or unnumbered address in the IPv4 case and the IPv6 link-local address on the interface in the IPv6 case.

The Hello message format is as follows:

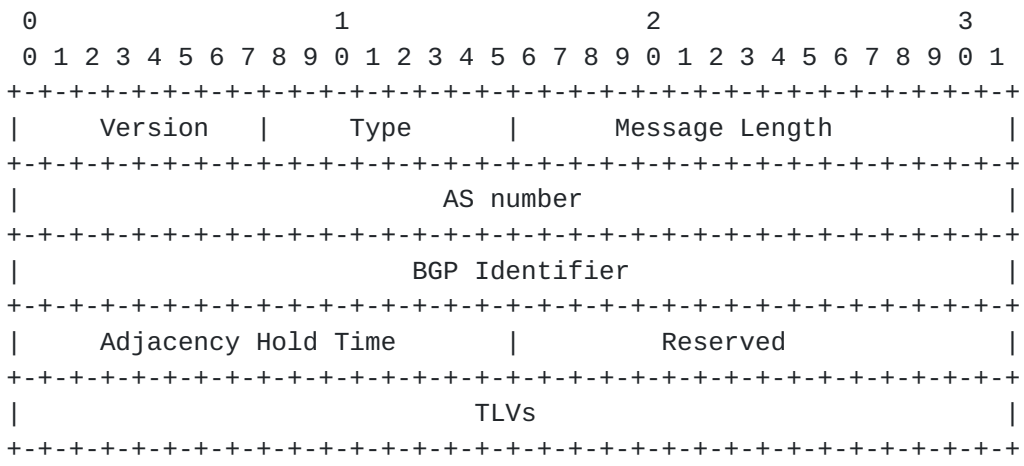

Figure 2: BGP Hello Message

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.

Adjacency Hold Time: Hello adjacency hold timer in seconds. Adjacency Hold Time specifies the time the receiving BGP neighbor router SHOULD maintain its neighbor adjacency state without receipt of another Hello. A value of 0 means that the receiving BGP peer should immediately mark that the sender is going down.

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.

BGP HELLO messages can be sent using either IPv4 or IPv6 addresses depending on the addressing used for session establishment and provisioned on the interfaces over which these messages are sent. Either IPv4 or IPv6 address (but never both on the same link) are used for the BGP Hello message exchange and the neighbor discovery mechanism based on the local configuration policy.

In a BGP DC network that is using IPv6 only in the fabric underlay, it is possible that no IPv6 global addresses are assigned to the interfaces between the nodes and the IPv6 Global address(es) are assigned only to the loopback interfaces of these nodes. Such a design could ease introducing of nodes in the fabric and links between them from a provisioning aspect. The BGP neighbor discovery mechanism described in this document works on links between routers having only IPv6 link-local addresses and setting up BGP sessions between them using their loopback IPv6 Global addresses in an automatic manner.

The neighbor discovery procedure using the Hello message is described in Section 7 and its relation with the BGP Keepalives and Hold Timer for the TCP session is described in Section 8.

6. Hello Message TLVs

The BGP Hello message carries TLVs as described in this section that enable exchange of information on a per interface basis between directly connected BGP neighbors. These messages enable the neighbor discovery process.

6.1. Accepted ASN List TLV

The Accepted ASN List TLV is an optional TLV that is used to signal the AS numbers from which the BGP router would accept BGP sessions. When not signaled, it indicates that the router will accept BGP peering from any ASN from its neighbors. Indicating the list of ASNs from which a router will accept BGP sessions helps avoid the neighbor discovery process getting stuck in a 1-way state where one side keeps attempting to setup adjacency while the other does not accept it due to incorrect ASN.

The operational and management aspects of this ASN based policy control for BGP neighbor discovery are described further in Section 10.

Only a single instance of this TLV is included and its format is shown below.

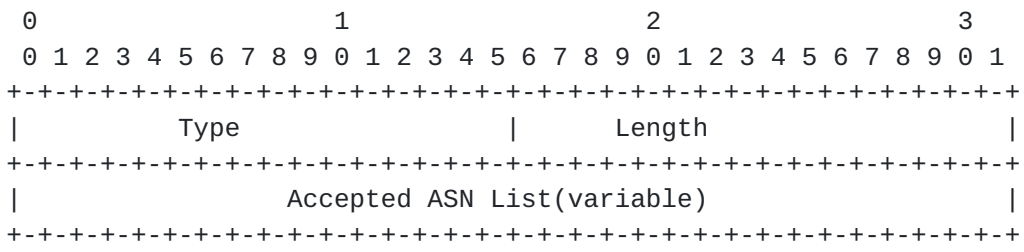

Figure 3: Accepted ASN List TLV

Type: TBD1

Length:Specifies the length of the Value field in octets (in multiple of 4)

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

6.2. Peering Address TLV

The Peering Address TLV is used to indicate to the neighbor the address to which they should establish the BGP TCP 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 done as part of the BGP TCP session establishment.

This is a mandatory TLV and at least one instance of this TLV MUST be present. Multiple instances of this TLV MAY be present one for each peering address (e.g. IPv4 and IPv6 or multiple IPv4 addresses for different AFI/SAFI sessions).

The Peering Address TLV format is shown below.

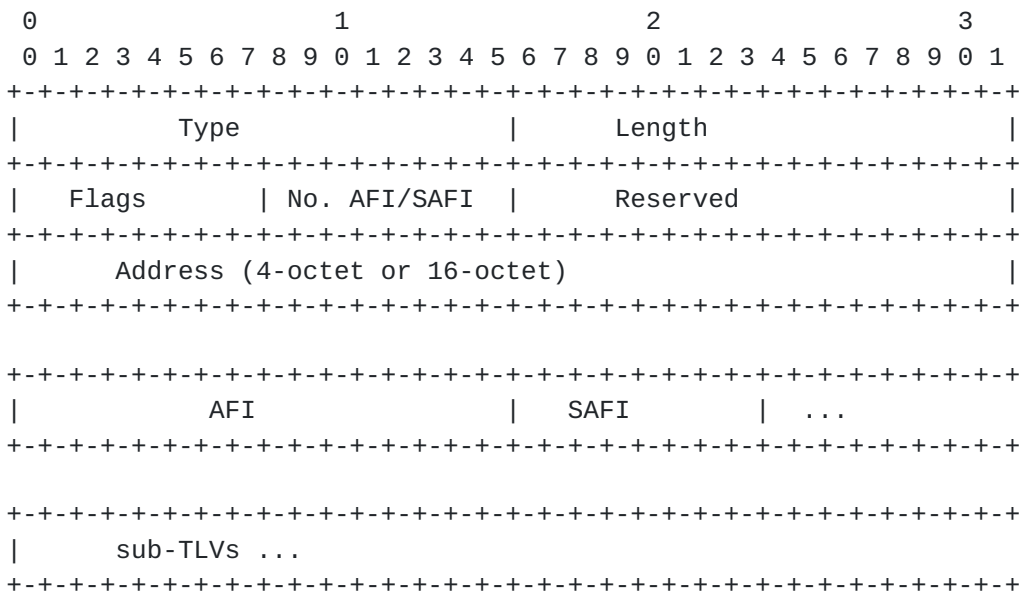

Figure 4: Peering Address TLV

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 octet 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

6.3. Local Prefix TLV

When the Peering Address to be used for the BGP TCP session establishment is not the directly connected interface address (e.g. when using loopback address) then local prefix(es) that cover its peering address(es) MUST be signaled by a BGP router to its neighbor

as part of the Hello message. 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 signaled. The Local Prefix TLV is this an optional TLV and it MUST be used to only signal prefixes that are locally configured on the router. The procedure for resolving the peering address signaled via the Peering Address TLV over the local prefixes signaled is described in Section 7.3.

The Local Prefix TLV format is as shown below.

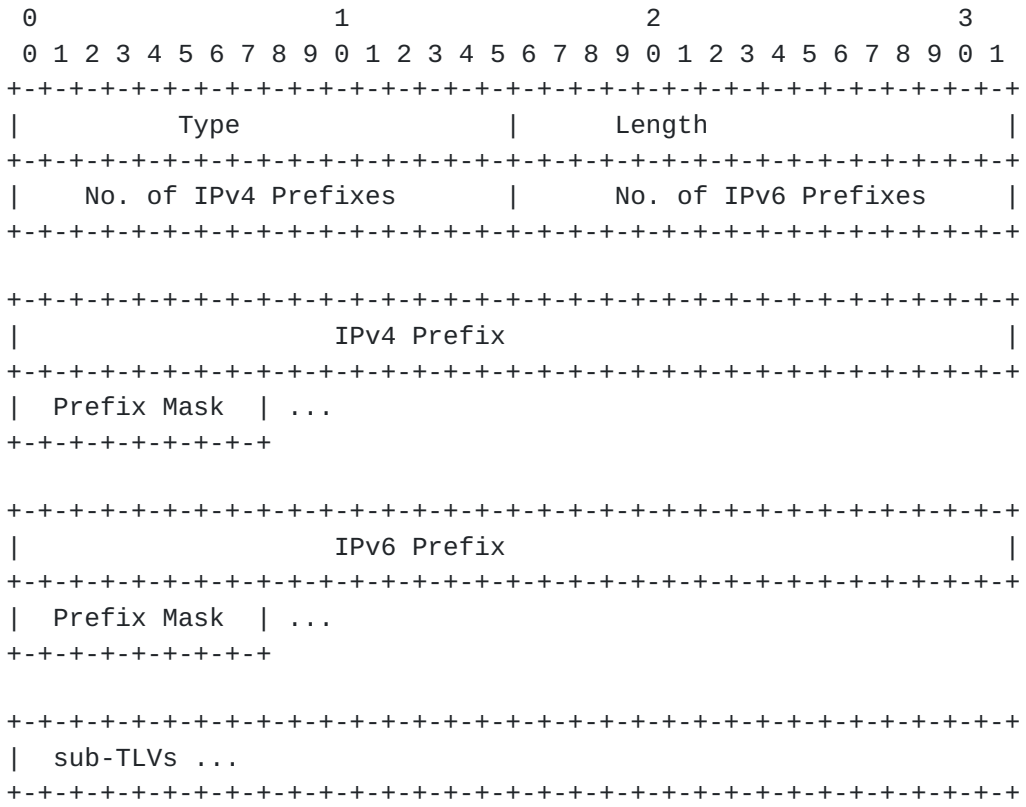

Figure 5: Local Prefix TLV

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

6.4. Link Attributes TLV

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. This TLV enables a BGP router to learn all its neighbors IP addresses on the specific link as well as its link identifiers. All the IPv4 addresses configured on the interface are signaled to the neighbor. When the interface has IPv4 unnumbered address then that is not included in this TLV. Only the IPv6 global addresses configured on the interface are signaled to the neighbor. In case of an interface running dual stack, both IPv4 and IPv6 addresses are signaled in a single TLV irrespective of which one is used for UDP message exchange.

More sub-TLVs may be defined in the future to exchange other link attributes between BGP neighbors.

The Link Attributes TLV format is as shown below.

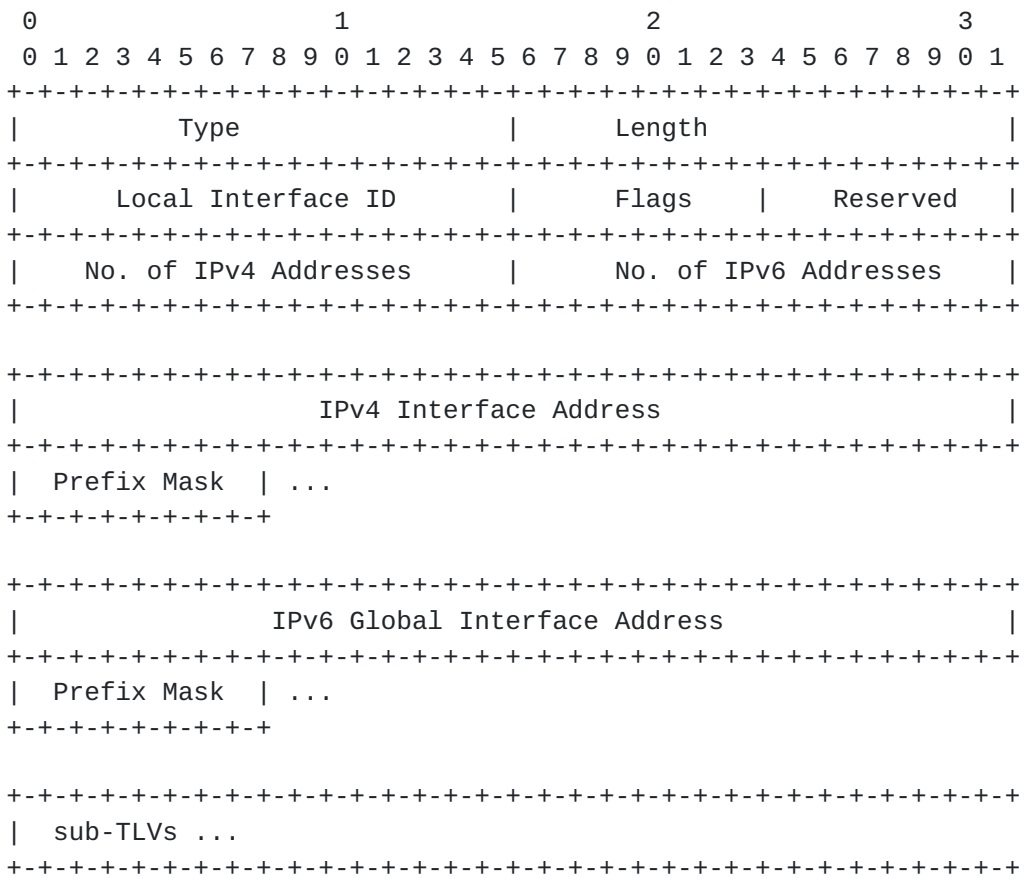

Figure 6: Link Attributes TLV

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). This helps uniquely identify the link even when there are multiple links between two neighbors using IPv4 unnumbered address or only having IPv6 link-local addresses.

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 addresses on the interface. When value is 0, then it indicates no IPv4 Prefixes are present or the interface is IPv4 unnumbered if it is enabled for IPv4

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 and the interface is only configured with IPv6 link-local addresses if it is enabled for IPv6.

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

6.5. Neighbor TLV

The Neighbor TLV is used by a BGP router to indicate its hello adjacency status with its neighboring router(s) on the specific link. The neighbor is identified by its Peering Address which has been accepted. The BGP TCP session establishment process begins when the hello adjacency is formed between the two neighbors over at least one directly connected link between them. Multiple instances of this TLV MAY be present in a Hello message - one for each peering address of each of its neighbor on that particular interface.

The Neighbor TLV format is as shown below.

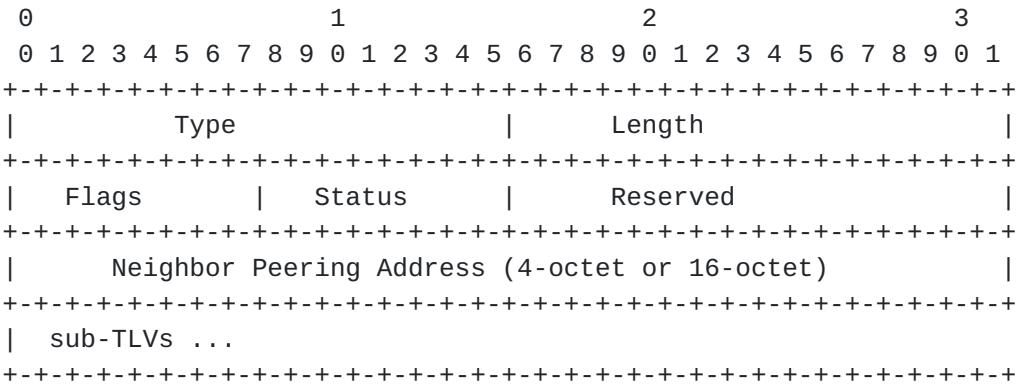

Figure 7: Neighbor TLV

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 TCP peering session has been established between the neighbors

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

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

Sub-TLVs : currently none defined

6.6. Cryptographic Authentication TLV

The Cryptographic Authentication TLV is an optional TLV that is used to introduce an authentication mechanism for BGP Hello message by securing against spoofing attacks. It also introduces a cryptographic sequence number carried in the Hello messages that can be used to protect against replay attacks. Using this Cryptographic Authentication TLV, one or more secret keys (with corresponding Security Association (SA) IDs) are configured on each BGP router. For each BGP Hello message, the key is used to generate and verify an HMAC Hash that is stored in the BGP Hello message. For the cryptographic hash function, this document proposes to use SHA-1, SHA-256, SHA-384, and SHA-512 defined in US NIST Secure Hash Standard (SHS) [FIPS-180-4]. The HMAC authentication mode defined in [RFC2104] is used. Of the above, implementations MUST include

support for at least HMAC-SHA-256, SHOULD include support for HMAC-SHA-1, and MAY include support for HMAC-SHA-384 and HMAC-SHA-512.

Further details for ensuring the security of the BGP Hello UDP messages are described in Section 9.

The Cryptographic Authentication TLV format is as shown below.

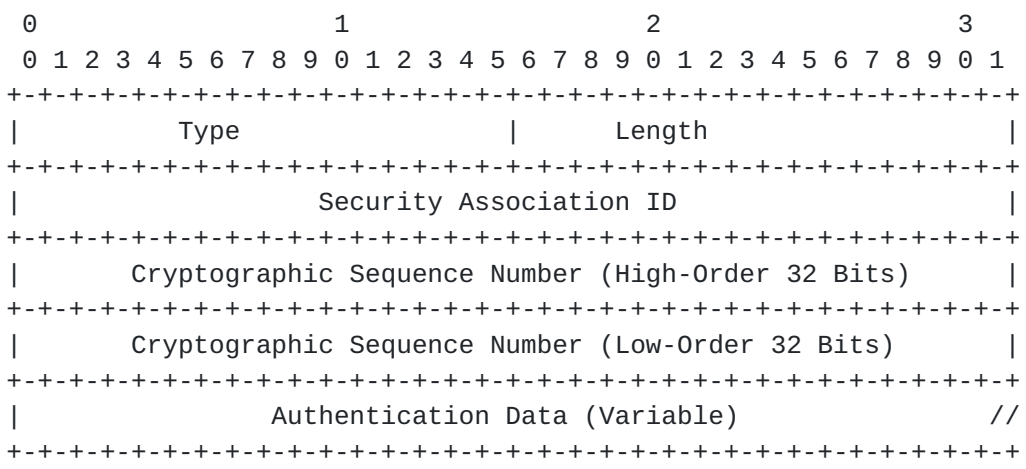

Figure 8: Cryptographic Authentication TLV

Type: TBD6

Length: Specifies the length of the Value field in octets

Security Association ID: The 32-bit field that maps to the authentication algorithm and the secret key used to create the message digest carried in Hello message payload.

Cryptographic Sequence Number: The 64-bit, strictly increasing sequence number that is used to guard against replay attacks. The 64-bit sequence number MUST be incremented for every BGP Hello message sent by the BGP router. Upon reception, the sequence number MUST be greater than the sequence number in the last BGP Hello message accepted from the sending BGP neighbor. Otherwise, the BGP hello message is considered a replayed packet and is dropped. The Cryptographic Sequence Number is a single space per BGP router.

Authentication Data: This field carries the digest computed by the Cryptographic Authentication algorithm in use. The length of the Authentication Data varies based on the cryptographic algorithm in use, which is shown below:

HMAC-SHA1 20 bytes

HMAC-SHA-256 32 bytes

HMAC-SHA-384 48 bytes

HMAC-SHA-512 64 bytes

7. Neighbor Discovery Procedure

The neighbor discovery mechanism in BGP is implemented with the introduction of an Interface state in BGP and an Adjacency Finite State Machine (FSM). This section describes the states, FSM and procedures involved.

7.1. Interface State

In order to perform neighbor discovery over its connected interfaces, BGP needs to maintain state for all its connected interfaces over which neighbor discovery is enabled. Once the neighbor discovery is enabled and the link is UP, then BGP starts sending its Hello messages with the TLVs listed in Section 6. The Neighbor TLV described in Section 6.5 is, however, not included until after a neighbor is learnt as part of the discovery process described in further sections.

These Hello messages are originated periodically at an interval which is less than or equal to one third of the Adjacency Hold Time specified in the message. The RECOMMENDED default value for the Adjacency Hold Time is 45 seconds and this makes the hello message interval to be 15 seconds. A Hello message SHOULD also be generated in a triggered manner during the neighbor discovery process as a change in the router's own or neighbor's Hello message is detected which results in change in adjacency state or parameters.

When a router does not receive a Hello message from its neighbor for a period equal to Adjacency Hold Time, then it MUST clean up its adjacency to this neighbor. The relationship of the Adjacency Hold Timer with the BGP Hold Timer at the TCP session level is described further in Section 8.

Before the interface is shut or the neighbor discovery is disabled on it, the router SHOULD attempt to send out triggered Hello messages with Adjacency Hold Time set to 0 and without including any Neighbor TLV in it to indicate that the neighbor discovery is being turned OFF on that router's interface. A router receiving a Hello message with Adjacency Hold Time set to 0 MUST clean up its adjacency to the originating router.

7.2. Adjacency State Machine

On a per interface basis, BGP needs to maintain an adjacency state for each neighbor that it discovers. The adjacency state is maintained as a FSM and it has the following states:

1. Init : This is the initial state that is setup when the router detects a hello message from a new neighbor that it has not seen previously. This is also the state to which the adjacency transitions to when the router no longer sees itself in a Neighbor TLV in the hello message from a neighbor.

2. 1-way : This is the state immediately after the Init when the router sends its Hello message with inclusion of the neighbor's Peering Address in a Neighbor TLV with the status set to 1-way.

3. Reject : This is the state (generally after Init) when the router detects that the neighbor cannot be accepted due to subnet mismatch on the addresses on either end of the link or a discrepancy in its Accepted ASN List TLV or due to some other local policy. The router then sends its Hello message with inclusion of this neighbor's Peering Address in a Neighbor TLV with the status set to rejection.

4. 2-way : This is the state after 1-way when the router detects its own Peering Address in a Neighbor TLV in the neighbor's hello message with the status set to 1-way or 2-way. It then updates the neighbor's status to 2-way in the Neighbor TLV in its own Hello message and sends it out. At this stage, both neighbors have accepted each other. On transition to this state, the router also installs peering route(s) in its own routing table corresponding to the prefix(es) received from the neighbor in its Local Prefix TLV so that reachability is established for the TCP session formation. Next the TCP session formation can be initialized via the BGP Peer FSM. If there is already a peering route to the same address on another interfaces, then this new interface is added as an ECMP path to it. If the BGP TCP session is already initialized (established or connection in progress) towards the same peering address then no further action is required on this BGP Peer FSM.

5. Established : This is the state after 2-way when the router has successfully setup its BGP TCP session with the neighbor's Peering Address. It then updates the neighbor's status to established in the Neighbor TLV in its own Hello message and sends it out.

Any downward transition from Established or 2-way state to a lower state results in removal of that interface from the peering route(s) for that neighbor and the deletion of the route itself when the last path is deleted. The deletion of the route may bring down the BGP TCP session.

A BGP TCP session with an auto-discovered neighbor may have one or more Hello adjacencies corresponding to it - one over each interconnecting link between them.

7.3. Peering Route

BGP auto-discovered neighbors MAY setup their BGP TCP session over a loopback address instead of using the directly connected interface address between them. When this is desired, the neighbors also advertise the loopback address host prefix (or optionally a prefix which covers more than a single loopback address when multiple are used for different peering sessions) in their Local Prefix TLV. Before the TCP session can be established, the reachability needs to be setup in both direction by each neighbor by programming their local prefixes in their forwarding plane. These routes that are programmed by BGP automatically using the prefixes advertised via the Local Prefix TLV are called Peering Routes.

Peering Routes serve two purposes. First, they enable reachability between the Peering Addresses (generally loopbacks) of the two neighbors so that the BGP TCP session may come up between them. Second, for the BGP routes learnt over the TCP session, where the next-hop is the neighbor, they also provide the BGP NH resolution.

Unlike other BGP routes, these are not recursive routes as in they point to the neighbor's interface and IP address. These routes that are setup as part of the neighbor discovery procedure are hence different from the regular iBGP and eBGP routes. These routes also MUST have a better administrative distance as compared to the iBGP and eBGP routes to ensure that they do not get displaced from the forwarding by BGP routes learnt over the same session that was established over these peering routes.

When there are multiple interconnecting links between two BGP neighbors, a single BGP TCP session may be setup between them over which routes are then exchanged. However, in the forwarding, the peering route will have multiple paths - one for each of these interconnecting links. So the BGP routes learnt over the session actually end up getting resolved over the peering route and in turn get the ECMP load balancing even with a single BGP session.

8. Interactions with Base BGP Protocol

The BGP Finite State Machine (FSM) as specified in [RFC4271] is unchanged and the BGP TCP session establishment, route updates and processing continues to follow the BGP protocol specifications.

BGP peering addresses along with their respective ASNs have traditionally been explicitly provisioned on both the BGP neighbors. The difference that neighbor discovery mechanism brings about is in elimination of this configuration as these parameters are learnt via the neighbor discovery procedure. Once BGP router learns its neighbor's peering address and ASN and has accepted it for peering based on its local policy configuration, then its initializes the BGP Peer FSM for this neighbor in the Idle State - just as if this neighbor was configured. From thereon, the BGP Peer FSM actions follows.

The BGP Keepalives and Hold Timer for the session over TCP apply unchanged and they govern the operations of the BGP TCP session and when it is brought down. While the BGP Keepalive works at the TCP session level, the BGP Adjacency Hold Timer monitors the liveliness on one or more underlying interconnecting link between the neighbors. The reachability for the BGP TCP session may be over more than one adjacency. The loss of BGP Hello messages on the UDP transport or some link failure can result in the expiry of the Adjacency Hold Timer. However, this does not result in bringing down of the BGP TCP session for an auto-discovered BGP neighbor by default. An implementation MAY provide an option to bring a BGP TCP session down when the Adjacency Hold Timer expiry brings down the last adjacency between neighbors very similar to how BFD down brings the session down.

When the BGP Peer FSM for an auto-discovered neighbor (i.e. one that is not provisioned explicitly), is in the Idle or Connect state then the adjacency state for that neighbor needs to be monitored to check if its BGP TCP session context needs to be cleaned-up. When there is no adjacency state for an auto-discovered neighbor in 2-way or Established state, then the BGP TCP session FSM state for such a neighbor MUST be cleaned-up when in Idle or Connect state. This is similar to when the configuration for a provisioned BGP neighbor is deleted from a BGP router.

Since the BGP neighbor discovery mechanism runs over a UDP socket, it is isolated from the core BGP protocol working which is TCP based. Implementations SHOULD ensure that the hello processing does not affect the base BGP operations and scalability. One option may be to run the BGP neighbor discovery mechanism in a separate thread from

the rest of BGP processing. These implementation details, however, are outside the scope of this document.

It is not generally expected that BGP sessions are explicitly provisioned along with the neighbor discovery mechanism. However, in such an event, the neighbor discovery mechanism MUST NOT affect or result in any changes to provisioned BGP neighbors and their operations. Specifically, BGP peering to auto-discovered neighbors MUST NOT be instantiated using the procedures described in this document when the same BGP neighbor is already provisioned. The configured BGP neighbor parameters take precedence and the auto-discovered values and parameters are not used for such configured BGP sessions.

Mechanisms like BFD monitoring and Fast External Failover that are currently used for eBGP sessions may still continue to be used where necessary and are not affected by the neighbor discovery mechanism.

9. Security Considerations

BGP routers accept TCP connection attempts to port 179 only from the provisioned BGP neighbors or, in some implementations, those from within a configured address range. With the BGP neighbor auto-discovery mechanism, it is now possible for BGP to automatically learn neighbors and initiate/receive TCP connections from them. This introduces the need for specific considerations to be taken care of to ensure security of the BGP protocol operations.

This document introduces UDP messages in BGP for the neighbor discovery mechanism using the BGP Hello messages. For security purposes, implementations MUST exchange the Hello messages only on interfaces specifically enabled for neighbor discovery. Hello messages MUST NOT be accepted on other than the 224.0.0.2 or FF02::2 addresses. Optionally, implementations MAY set TTL to 255 when originating the Hello messages and receivers check specifically for the TLV to be 254 and discard the packet when this is not the case. This ensures that the Hello packets signaling happens between directly connected BGP routers only.

The BGP neighbor discovery mechanism is expected to be run typically in DCs and between physically connected routers that are trustworthy. The Cryptographic Authentication TLV (as described in Section 6.6) SHOULD be used in deployments where this assumption of trustworthiness is not valid. This mechanism is similar to one defined for LDP Hello messages that are also UDP based as specified in [RFC7349]. An updated future version of this document will describe similar procedures for BGP hello in more details.

Once the BGP hello messages and the neighbor discovery mechanism is secured, then the security considerations for BGP protocol operations apply for the auto-discovered neighbor sessions. Specifically, for the BGP TCP sessions with the automatically discovered directly connected neighbors, the TTL of the BGP TCP messages (dest port=179) MUST be set to 255. Any received BGP TCP message with TTL being less than 254 MUST be dropped according to [RFC5082].

10. Manageability Considerations

This section is structured as recommended in [RFC5706].

10.1. Operational Considerations

The BGP neighbor discovery mechanism introduced by this document is not applicable to general BGP deployments and is specifically meant for DC networks where BGP is used as a hop-by-hop routing protocol as described in [RFC7938]. The neighbor discovery mechanism hence SHOULD NOT be enabled by default in BGP.

Implementations SHOULD provide configuration methods that allow enablement of BGP neighbor discovery on specific local interfaces. In a DC network, it is expected that the operator selects the appropriate links on which to enable this e.g. on a Tier 2 node it is enabled on all links towards the Tier 1 and Tier 3 nodes while on a Tier 3 node, it may be only enabled on the links towards the Tier 2 node. The details of this enablement are outside the scope of this document since it varies based on the DC design and may be implementation specific.

Implementations SHOULD provide configuration methods that enable the setup of BGP neighbor templates that enables operator to setup BGP neighbor discovery parameters on the BGP router. Some of the aspects to be considered in such a template are:

- Local address to be used for the BGP TCP session peering along with the local ASN and the AFI/SAFI enabled for the auto-discovered sessions

- BGP policies to be enabled for the auto-discovered sessions

- Optionally specify the list of ASNs with which auto-discovered sessions should be brought up. This is to ensure that when links between different Tier nodes are not used by BGP when they get connected wrongly due to accidents (e.g. say a Tier 3 node is connected to a Tier 1 node).

- Authentication methods that are need to be enabled in an environment which is not secure

- Local interfaces over which the specific template needs to be applied for BGP neighbor discovery

- Other parameters like the Adjacency Hold Timer value to be used or other optional features

This mechanism does not impose any restrictions on the way ASNs or addresses are assigned to the nodes. Various automatic provisioning, auto-configuration or zero-touch-provisioning mechanisms may be used.

Implementations SHOULD report the state of the BGP operations over each link enabled for neighbor discovery including the status of all adjacencies learnt over it. Implementations SHOULD also report the operations of the auto-discovered BGP TCP peering sessions similar to the provisioned BGP neighbors.

Implementations SHOULD support logging of events like discovery of an adjacency using neighbor discovery including peering route updates and events like triggering of BGP TCP session establishment for them. Errors and alarms related to loss of adjacencies and tear down of BGP TCP peering sessions SHOULD also be generated so they could be monitored.

10.2. Management Considerations

This document introduces UDP based messaging in BGP protocol and therefore the necessary fault management mechanisms are required to be implemented for the same. Implementations MUST discard unsupported message types or version types other than 4 received over a UDP session. Such messages MUST NOT affect the neighbor discovery mechanism in operation using the Hello messages. Unknown TLVs received via the Hello messages MUST be ignored and the rest of the Hello message MUST be processed. Implementations SHOULD discard Hello messages with malformed TLVs and this should be logged as an error.

11. IANA Considerations

This documents requests IANA for updates to the BGP Parameters registry as described in this section.

11.1. 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: 179 (preferred value) -- To be assigned by IANA.		

11.2. 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 Prefix	This document
4	Link Attributes	This document
5	Neighbor	This document
6	Cryptographic Authentication	This document
7-65500	Unassigned	
65501-65534	Experimental	This document
65535	Reserved	This document

12. Acknowledgements

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

13. Contributors

Satya Mohanty
Cisco
Email: satyamoh@cisco.com

Shunwan Zhuang
Huawei
Email: zhuangshunwan@huawei.com

Chao Huang
Alibaba Inc
Email: jingtan.hc@alibaba-inc.com

Guixin Bao
Alibaba Inc
Email: guixin.bgx@alibaba-inc.com

Jinghui Liu
Ruijie Networks
Email: liujh@ruijie.com.cn

Zhichun Jiang
Tencent
Email: zcjiang@tencent.com

Shaowen Ma
Juniper Networks
mashaowen@gmail.com

14. References

14.1. 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>.

14.2. Informative References

[FIPS-180-4]
"Secure Hash Standard (SHS), FIPS PUB 180-4", March 2012.

[I-D.ietf-lsvr-bgp-spf]
Patel, K., Lindem, A., Zandi, S., and W. Henderickx, "Shortest Path Routing Extensions for BGP Protocol", draft-ietf-lsvr-bgp-spf-01 (work in progress), May 2018.

[I-D.ketant-idr-bgp-ls-bgp-only-fabric]
Talaulikar, K., Filsfils, C., ananthamurthy, k., and S. Zandi, "BGP Link-State Extensions for BGP-only Fabric", draft-ketant-idr-bgp-ls-bgp-only-fabric-00 (work in progress), March 2018.

[RFC2104] Krawczyk, H., Bellare, M., and R. Canetti, "HMAC: Keyed-Hashing for Message Authentication", RFC 2104, DOI 10.17487/RFC2104, February 1997, <https://www.rfc-editor.org/info/rfc2104>.

[RFC5706] Harrington, D., "Guidelines for Considering Operations and Management of New Protocols and Protocol Extensions", RFC 5706, DOI 10.17487/RFC5706, November 2009, <https://www.rfc-editor.org/info/rfc5706>.

[RFC7349] Zheng, L., Chen, M., and M. Bhatia, "LDP Hello Cryptographic Authentication", RFC 7349, DOI 10.17487/RFC7349, August 2014, <https://www.rfc-editor.org/info/rfc7349>.

[RFC7752] Gredler, H., Ed., Medved, J., Previdi, S., Farrel, A., and S. Ray, "North-Bound Distribution of Link-State and Traffic Engineering (TE) Information Using BGP", RFC 7752, DOI 10.17487/RFC7752, March 2016, <https://www.rfc-editor.org/info/rfc7752>.

[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

Ketan Talaulikar
Cisco Systems

Email: ketant@cisco.com

Kunyang Bi
Huawei

Email: bikunyang@huawei.com

Jeff Tantsura
Nuage Networks

Email: jefftant.ietf@gmail.com

Nikos Triantafillis

Email: ntriantafillis@gmail.com