Network Working Group
Internet Draft
Intended status: Standard
Expires: September 14, 2023

L. Dunbar
Futurewei
S. Hares
Hickory Hill Consulting
R. Raszuk
Arrcus
K. Majumdar
Microsoft
Gyan Mishra
Verizon
March 14, 2023

BGP UPDATE for SDWAN Edge Discovery
draft-ietf-idr-sdwan-edge-discovery-07

Abstract

The document describes the encoding of BGP UPDATE messages for the SDWAN edge node property discovery.

In the context of this document, BGP Route Reflector (RR) is the component of the SDWAN Controller that receives the BGP UPDATE from SDWAN edges and in turns propagates the information to the intended peers that are authorized to communicate via the SDWAN overlay network.

Status of this Memo

Copyright Notice

Table of Contents

1. Introduction

[SDWAN-BGP-USAGE] illustrates how BGP [RFC4271] is used as a control plane for a SDWAN network. SDWAN network refers to a policy-driven network over multiple heterogeneous underlay networks to get better WAN bandwidth management, visibility, and control.

The document describes BGP UPDATE messages for an SDWAN edge node to advertise its properties to its RR which then propagates that information to the authorized peers.

2. Conventions used in this document

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

The following acronyms and terms are used in this document:

Cloud DC: Off-Premise Data Centers that usually host applications and workload owned by different organizations or tenants.

Controller: Used interchangeably with SDWAN controller to manage SDWAN overlay path creation/deletion and monitor the path conditions between sites.

CPE: Customer (Edge) Premises Equipment.

CPE-Based VPN: Virtual Private Secure network formed among CPEs. This is to differentiate such VPNs from most commonly used PE-based VPNs discussed in [RFC4364].

MP-NLRI: Multi-Protocol Network Layer Reachability Information [MP_REACH_NLRI] Path Attribute defined in RFC4760.

SDWAN End-point: can be the SDWAN edge node address, a WAN port address (logical or physical) of a SDWAN edge node, or a client port address.

OnPrem: On Premises data centers and branch offices.

RR Route Reflector.

SDWAN: Software Defined Wide Area Network. In this document, "SDWAN" refers to policy-driven transporting IP packets over multiple different underlay networks to get better WAN bandwidth management, visibility and control.

SDWAN Segmentation: Segmentation is the process of dividing the network into logical sub-networks.

SDWAN VPN: refers to the Client's VPN, which is like the VRF on the PEs of a MPLS VPN. One SDWAN client VPN can be mapped one or multiple SD-WAN virtual topologies. How Client VPN is mapped to a SDWAN virtual topology is governed by policies.

SDWAN Virtual Topology: Since SDWAN can connect any nodes, whereas MPLS VPN connects a fixed number of PEs, one SDWAN Virtual Topology refers to a set of edge nodes and the tunnels (including both IPsec tunnels and/or MPLS tunnels) interconnecting those edge nodes.

VPN Virtual Private Network.

VRF VPN Routing and Forwarding instance.

WAN Wide Area Network.

3. Framework of SDWAN Edge Discovery

3.1. The Objectives of SDWAN Edge Discovery

The objectives of SDWAN edge discovery are for an SDWAN edge node to discover its authorized peers and their associated properties to establish secure overlay tunnels. The attributes to be propagated includes:

- the SDWAN (client) VPNs information,
- the attached routes under the SDWAN VPNs,
- the properties of the underlay networks over which the client routes can be carried, and potentially more.

Some SDWAN peers are connected by both trusted VPNs and untrusted public networks. Some SDWAN peers are connected only by untrusted public networks. For the traffic over untrusted networks, IPsec Security Associations (IPsec SA) must be established and maintained. If an edge node has network ports behind a NAT, the NAT properties need to be discovered by the authorized SDWAN peers.

Like any VPN networks, the attached client's routes belonging to specific SDWAN VPNs can only be exchanged with the SDWAN peer nodes authorized to communicate.

3.2. Comparing with Pure IPsec VPN

A pure IPsec VPN has IPsec tunnels connecting all edge nodes over public networks. Therefore, it requires stringent authentication and authorization (i.e., IKE Phase 1) before other properties of IPsec

SA can be exchanged. The IPsec Security Association (SA) between two untrusted nodes typically requires the following configurations and message exchanges:

- IPsec IKEv2 to authenticate with each other.
- Establish IPsec SA
    - o Local key configuration
    - o Remote Peer address (192.10.0.10<->172.0.01)
    - o IKEv2 Proposal directly sent to peer.
        - o Encryption method, Integrity sha512
    - o Transform set.
- Attached client prefixes discovery.
    - o By running routing protocol within each IPsec SA
    - o If multiple IPsec SAs between two peer nodes are established to achieve load sharing, each IPsec tunnel needs to run its own routing protocol to exchange client routes attached to the edges.
- Access List or Traffic Selector
    - o Permit Local-IP1, Remote-IP2

In a BGP-controlled SDWAN network over hybrid MPLS VPN and public internet underlay networks, all edge nodes and RRs are already connected by private secure paths. The RRs have the policies to manage the authentication of all peer nodes. More importantly, when an edge node needs to establish multiple IPsec tunnels to many edge nodes, all the management information can be multiplexed into the secure management tunnel between RR and the edge node. Therefore, the amount of authentication in a BGP-Controlled SDWAN network can be significantly reduced.

Client VPNs are configured via VRFs, just like the configuration of the existing MPLS VPN. The IPsec equivalent traffic selectors for local and remote routes are achieved by importing/exporting VPN Route Targets. The binding of client routes to IPsec SA is dictated by policies. As a result, the IPsec configuration for a BGP controlled SDWAN (with mixed MPLS VPN) can be simplified:

- The SDWAN controller has the authority to authenticate edges and peers. Remote Peer association is controlled by the SDWAN Controller (RR)
- The IKEv2 proposals, including the IPsec Transform set, can be sent directly to peers, or incorporated in a BGP UPDATE.

- BGP UPDATE: Announces the client route reachability for all permitted parallel tunnels/paths.
  - o There is no need to run multiple routing protocols in each IPsec tunnel.
- Importing/exporting Route Targets under each client VPN (VRF) achieves the traffic selection (or permission) among clients' routes attached to multiple edge nodes.

3.3. Client Route UPDATE and SDWAN Tunnel UPDATE

As described in [SDWAN-BGP-USAGE], two separate BGP UPDATE messages are used for SDWAN Edge Discovery:

- Client routes BGP UPDATE:
  This UPDATE is precisely the same as the BGP VPN client route UPDATE. It uses the Encapsulation Extended Community and the Color Extended Community to link with the SDWAN Tunnels UPDATE Message as specified in section 8 of [RFC9012].

  A new Tunnel Type (SDWAN-Hybrid) is added and used by the Encapsulation Extended Community or the Tunnel-Encap Path Attribute [RFC9012] to indicate mixed underlay networks.

- SDWAN UPDATE.
  This UPDATE is for an edge node to advertise the properties of directly attached underlay networks, including the NAT information, pre-configured IPsec SA identifiers, and/or the underlay network specific information. This UPDATE can also include the detailed IPsec SA attributes, such as keys, nonce, encryption algorithms, etc.

In the following figure, there are potentially four underlay paths between C-PE1 and C-PE2, even though C-PE1/C-PE2 might not use all four underlay paths:

a) MPLS-in-GRE path.

b) node-based IPsec tunnel [2.2.2.2<->1.1.1.1]. As C-PE2 has two public internet facing WAN ports, either of those two WAN port IP addresses can be the outer destination address of the IPsec encapsulated data packets.

c) port-based IPsec tunnel [192.0.0.1 <-> 192.10.0.10]; and

d) port-based IPsec tunnel [172.0.0.1 <-> 160.0.0.1].

```
                                       +---+
                        +--------------|RR |----------+
                       /  Untrusted    +-+-+           \
                      /                                 \
                     /                                   \
             +---------+  MPLS Path                      +-------+
     11.1.1.x| C-PE1   A1-------------------------------B1 C-PE2 |10.1.1.x
             |         |                                 |       |
     21.1.1.x|         A2(192.10.0.10)------( 192.0.0.1)B2       |20.1.1.x
             |         |                                 |       |
             | Addr    A3(160.0.0.1) --------(170.0.0.1)B3 Addr  |
             | 1.1.1.1 |                                 |2.2.2.2|
             +---------+                                 +-------+
```

Figure 1: Hybrid SDWAN

C-PE2 advertises the attached client routes as below:

Client Route UDPATE:

Extended community: RT for SDWAN VPN 1
NLRI: AFI=IPv4/IPv6 & SAFI = VPN
  Prefix: 10.1.1.x; 20.1.1.x
  NextHop: 2.2.2.2 (C-PE2)
Encapsulation Extended Community: tunnel-type=SDWAN-Hybrid
Color Extended Community: Site-identifier

The Client Route UPDATE is recursively resolved to the SDWAN UPDATE which specifies the detailed properties including IPsec properties of hybrid WAN underlay tunnels terminated at the C-PE2:

SDWAN UPDATE:

C-PE2 can use the following Update messages to advertise the properties of Internet facing ports 192.0.0.1 & 170.0.0.1, and their associated IPsec SA related parameters.

Update #1 for the properties associated with the WAN port 192.0.0.1, such as the NAT properties, the underlay network properties, etc. [Details in Section 9.1]

Update #2 for the properties associated with the WAN port 170.0.0.1 associated properties. [Details in Section 9.1]

Update #3 for IPsec parameters associated with IPsec tunnel terminated at the Node level (2.2.2.2), such as the supported encryption methods, public keys, etc. [Details in Section 9.2].

3.4. Edge Node Discovery

The basic scheme of SDWAN edge node discovery using BGP consists of the following:

- Secure connection to a SDWAN controller (i.e., RR in this context):
  For an SDWAN edge with both MPLS and IPsec paths, the edge node should already have a secure connection to its controller, i.e., RR in this context. For an SDWAN edge that is only accessible via Internet, the SDWAN edge, upon power-up, establishes a secure tunnel (such as TLS or SSL) with the SDWAN central controller whose address is preconfigured on the edge node. The central controller informs the edge node of its local RR. The edge node then establishes a transport layer secure session with the RR (such as TLS or SSL).

- The Edge node will advertise its own properties to its designated RR via the secure connection.

- The RR propagates the received information to the authorized peers.

- The authorized peers can establish the secure data channels (IPsec) and exchange more information among each other.

For an SDWAN deployment with multiple RRs, it is assumed that there are secure connections among those RRs. How secure connections are established among those RRs is out of the scope of this document. The existing BGP UPDATE propagation mechanisms control the edge properties propagation among the RRs.

For some environments where the communication to RR is highly secured, [RFC9016] IKE-less can be deployed to simplify IPsec SA establishment among edge nodes.

4. Constrained propagation of BGP UPDATE

4.1. SDWAN Segmentation, SDWAN Virtual Topology and Client VPN

In SDWAN deployment, "SDWAN Segmentation" is a frequently used term, referring to partitioning a network into multiple sub-networks, just like MPLS VPNs. "SDWAN Segmentation" is achieved by creating SDWAN virtual topologies and SDWAN VPNs. An SDWAN virtual topology consists of a set of edge nodes and the tunnels (a.k.a. underlay paths), including both IPsec tunnels and/or MPLS VPN tunnels, interconnecting those edge nodes.

An SDWAN VPN is configured in the same way as the VRFs of an MPLS VPN. One SDWAN client VPN can be mapped to multiple SD-WAN virtual topologies. SDWAN Controller governs the policies of mapping a client VPN to SDWAN virtual topologies.

Each SDWAN edge node may need to support multiple VPNs. Route Target is used to differentiate the SDWAN VPNs. For example, in the picture below, the "Payment-Flow" on C-PE2 is only mapped to the virtual topology of C-PEs to/from Payment Gateway, whereas other flows can be mapped to a multipoint-to-multipoint virtual topology.

```
                                       +---+
                        +--------------|RR |----------+
                       /  Untrusted    +-+-+           \
                      /                                 \
                     /                                   \
             +---------+  MPLS Path                       +-------+
     11.1.1.x| C-PE1   A1-------------------------------B1 C-PE2 |10.1.1.x
             |         |                                 |       |
     21.1.1.x|         A2(192.10.0.10)------( 192.0.0.1)B2       |20.1.1.x
             |         |                                 |       |
             | Addr    A3(160.0.0.1) --------(170.0.0.1)B3 Addr  |11.2.2.x
             | 1.1.1.1 |                              /  |2.2.2.2|
             +---------+                             /   +-------+
                        \                           /
                         \                         /PaymentFlow
                          \                       /
                           \                +----+----+
                            +---------------| payment |
                                            | Gateway |
                                            +---------+
```

Figure 2: SDWAN Virtual Topology & VPN

4.2. Constrained Propagation of Edge Capability

BGP has a built-in mechanism [RFC4684] to dynamically achieve the constrained distribution of edge information. In a nutshell, an SDWAN edge sends RT Constraint (RTC) NLRI to the RR for the RR to install an outbound route filter, as shown in the figure below:

```
         RT Constraint                   RT constraint
         NLRI={SDWAN#1, SDWAN#2}         NLRI={SDWAN#1, SDWAN#3}
                 ----->                 +---+      <-----------
                   +--------------------|RR1|------------------+
                   | Outbound Filter    +---+  Outbound Filter |
                   | Permit SDWAN#1,#2        Permit SDWAN#1,#3|
                   | Deny all                 Deny all         |
                   |   <-------                --------->      |
                   |                                           |
             +-----+---+  MPLS Path                      +-----+-+
     11.1.1.x| C-PE1   A1-------------------------------B1 C-PE2 |10.1.1.x
             |         |                                 |       |
     21.1.1.x|         A2(192.10.0.10)------( 192.0.0.1)B2       |20.1.1.x
             |         |                                 |       |
             | Addr    A3(160.0.0.1) --------(170.0.0.1)B3 Addr  |
             | 1.1.1.1 |                                 |2.2.2.2|
             +---------+                                 +-------+
     SDWAN VPN #1                                          SDWAN VPN #1
     SDWAN VPN #2                                          SDWAN VPN #3
```

Figure 3: Constraint propagation of Edge Property

However, a SDWAN overlay network can span across untrusted networks, RR can't trust the RT Constraint (RTC) NLRI BGP UPDATE from any nodes. RR can only process the RTC NLRI from authorized peers for a SDWAN VPN.

It is out of the scope of this document on how RR is configured with the policies to filter out unauthorized nodes for specific SDWAN VPNs.

When the RR receives BGP UPDATE from an edge node, it propagates the received UPDATE message to the nodes that are in the Outbound Route filter for the specific SDWAN VPN.

5. Client Route UPDATE

The SDWAN network's Client Route UPDATE message is the same as the L3 VPN or EVPN client route UDPATE message. The SDWAN Client Route UPDATE message uses the Encapsulation Extended Community and the Color Extended Community to link with the SDWAN Underlay UPDATE Message.

5.1. SDWAN VPN ID in Client Route Update

An SDWAN VPN is same as a client VPN in a BGP controlled SDWAN network. The Route Target Extended Community should be included in a Client Route UPDATE message to differentiate the client routes from routes belonging to other VPNs.

5.2. SDWAN VPN ID in Data Plane

For an SDWAN edge node which can be reached by both MPLS and IPsec paths, the client packets reached by MPLS network will be encoded with the MPLS Labels based on the scheme specified by [RFC8277].

For GRE Encapsulation within an IPsec tunnel, the GRE key field can be used to carry the SDWAN VPN ID. For network virtual overlay (VxLAN, GENEVE, etc.) encapsulation within the IPsec tunnel, the Virtual Network Identifier (VNI) field is used to carry the SDWAN VPN ID.

6. SDWAN Underlay UPDATE

The hybrid underlay tunnel UPDATE is to advertise the detailed properties associated with the public facing WAN ports and IPsec tunnels.

6.1. NLRI for SDWAN Underlay Tunnel Update

A new NLRI (SDWAN-SAFI=74) is introduced within the MP_REACH_NLRI Path Attribute of RFC4760, for advertising the detailed properties of the SDWAN tunnels terminated at the edge node:

```
+------------------+
|    Route Type    | 2 octet
+------------------+
|     Length       | 2 Octet
+------------------+
|   Type Specific  |
~ Value (Variable) ~
|                  |
+------------------+
```

where:

- Route-Type: 2 octet value to define the encoding of the rest of the SDWAN the NLRI.
- Length: 2 octets of length expressed in bits as defined in [RFC4760].

This document defines the following SDWAN Route types:

- Route-Type = 1: For advertising the detailed properties of the SDWAN tunnels terminated at the edge, where the transport network port can be uniquely identified by a tuple of three values <Port-Local-ID>, SDWAN-Color/Site-ID, SDWAN-Node-ID>. The SDWAN NLRI Route-Type =1 has the following encoding:

```
                +-------------------+
                |  Route Type = 1   | 2 octet
                +-------------------+
                |     Length        | 2 Octet
                +-------------------+
                |   Port-Local-ID   | 4 octets
                +-------------------+
                |SDWAN-Color/SiteID| 4 octets
                +-------------------+
                |  SDWAN-Node-ID    | 4 or 16 octets
                +-------------------+
```

  o Port local ID: SDWAN edge node Port identifier, which is locally significant. If the SDWAN NLRI applies to multiple WAN ports, this field is NULL.
  o SDWAN-Color/SiteID: to correlate with the Color-Extended-community included in the client routes UPDATE. When a client route can be reached by multiple SDWAN edges co-located at one site, the SDWAN-Site-Color can indicate the site identifier.
  o SDWAN Node ID: The node's IPv4 or IPv6 address.
- Route-Type = others: for supporting various other SDWAN applications, which will be defined later.

6.2. SDWAN-Hybrid Tunnel Encoding

A new BGP Tunnel-Type=SDWAN-Hybrid (code point 25) is to indicate hybrid underlay tunnels.

```
    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
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   | Tunnel-Type=25(SDWAN-Hybrid ) | Length (2 Octets)             |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                           Sub-TLVs                            |
   |                                                               |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
                     SDWAN Hybrid Value Field
```

6.3. IPsec-SA-ID Sub-TLV

IPsec-SA-ID Sub-TLV within the Hybrid Underlay Tunnel UPDATE indicates one or more preestablished IPsec SAs by using their identifiers, instead of listing all the detailed attributes of the IPsec SAs.

Using an IPsec-SA-ID Sub-TLV not only greatly reduces the size of BGP UPDATE messages, but also allows the pairwise IPsec rekeying process to be performed independently.

The following is the structure of the IPsec-SA-ID sub-TLV:

```
    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=64 (IPsec-SA-ID subTLV)   |  Length (2 Octets)            |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                      IPsec SA Identifier #1                   |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                      IPsec SA Identifier #2                   |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

If the client traffic needs to be encapsulated in a specific way within the IPsec ESP Tunnel, such as GRE or VxLAN, etc., the corresponding Tunnel-Encap Sub-TLV needs to be prepended right before the IPsec-SA-ID Sub-TLV.

6.4. Extended Port Attribute Sub-TLV

Extended Port Attribute Sub-TLV is to advertise the properties associated with a public internet facing WAN port which might be behind NAT. An SDWAN edge node can query a STUN Server (Session

Traversal of UDP through Network address translation [RFC3489]) to get the NAT properties, including the public IP address and the Public Port number, to pass to its peers.

The location of a NAT device can be:

- Only the initiator is behind a NAT device. Multiple initiators can be behind separate NAT devices. Initiators can also connect to the responder through multiple NAT devices.
- Only the responder is behind a NAT device.
- Both the initiator and the responder are behind a NAT device.

The initiator's address and/or responder's address can be dynamically assigned by an ISP or when their connection crosses a dynamic NAT device that allocates addresses from a dynamic address pool.

As one SDWAN edge can connect to multiple peers, the pair-wise NAT exchange as IPsec's IKE is not efficient. In the BGP Controlled SDWAN, NAT properties for a WAN port are encoded in the Extended Port Attribute sub-TLV, which the following format:

```
 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=65(extPort|  EncapExt subTLV Length       |I|O|R|R|R|R|R|R|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| NAT Type      |  Encap-Type   |Trans networkID|     RD ID     |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                  Local  IP Address                            |
           32-bits for IPv4, 128-bits for Ipv6
                   ~~~~~~~~~~~~
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                  Local  Port                                  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                Public IP                                      |
           32-bits for IPv4, 128-bits for Ipv6
                   ~~~~~~~~~~~~
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                Public Port                                    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|               Extended SubSub-TLV                             |
~                                                               ~
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Where:

- o Extended Port Attribute Type (=65): indicating it is the Extended Port Attribute SubTLV.
- o PortExt subTLV Length: the length of the subTLV.
- o Flags:
  - - I bit (CPE port address or Inner address scheme)
    If set to 0, indicate the inner (private) address is IPv4.
    If set to 1, it indicates the inner address is IPv6.

  - - O bit (Outer address scheme):
    If set to 0, indicate the public (outer) address is IPv4.
    If set to 1, it indicates the public (outer) address is IPv6.

  - - R bits: reserved for future use. Must be set to 0 now.

- o NAT Type.the NAT type can be: without NAT; 1:1 static NAT; Full Cone; Restricted Cone; Port Restricted Cone; Symmetric; or Unknown (i.e. no response from the STUN server).
- o Encap Type.the encapsulation types supported for the port facing public network, such as IPsec+GRE, IPsec+VxLAN, IPsec without GRE, GRE (when packets don't need encryption)
- o Transport Network ID.Central Controller assign a global unique ID to each transport network.
- o RD ID.Routing Domain ID.need to be global unique.
- o Local IP.The local (or private) IP address of the WAN port.
- o Local Port.used by Remote SDWAN edge node for establishing IPsec to this specific port.
- o Public IP.The IP address after the NAT. If NAT is not used, this field is set to NULL.
- o Public Port.The Port after the NAT. If NAT is not used, this field is set to NULL.
- o Extended SubSub-TLV: for carrying additional information about the underlay networks.

6.5. Extended SubSub-TLV

Two types of the Extended SubSub-TLVs are specified in this document: Underlay Network Transport SubSub-TLV and the underlay Geo Location SubSub-TLV".

6.5.1. Underlay Network Transport SubSub-TLV

The Underlay Network Transport SubSub-TLV is an optional Sub-TLV to carry the WAN port connection types and bandwidth, such as LTE, DSL, Ethernet, etc.

The format of this Sub-TLV is as follows:

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| UnderlayType  |      Length   |      Flag     |    Reserved   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|Connection Type|   Port Type   |        Port Speed             |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Where:

Underlay Network Properties sub Type=66.

Length: always 6 bytes.

Flag: a 1 octet value.

Reserved: 1 octet of reserved bits. It SHOULD be set to zero on transmission and MUST be ignored on receipt.

Connection Type: are listed below as:

Wired - 1
WIFI - 2
LTE - 3
5G - 4

Port Type: There are different types of ports. They are listed Below as:

Ethernet - 1
Fiber Cable - 2
Coax Cable - 3
Cellular - 4

Port Speed: The port seed is defined as 2 octet value. The values are defined as Gigabit speed.

6.5.2. Geo Location SubSub-TLV

For a large SDWAN heterogeneous deployment where SDWAN Node-ID is not enough to identify the exact location of an SDWAN edge, [LISP-GEOLoc] sub-TLV can be appended to the Extended Port Attribute Sub-TLV to describe the accurate location of the transport network node.

7. IPsec SA Property Sub-TLVs

This section describes the detailed IPsec SA properties sub-TLVs. When the IPsec SA properties are associated with the node, any of the node's WAN ports can be the outer destination address of the IPsec encapsulated data packets.

7.1. IPsec SA Nonce Sub-TLV

The Nonce Sub-TLV is based on the Base DIM sub-TLV as described the Section 10.1 of [SECURE-EVPN]. The following fields are removed because:

- the Originator ID is same as the Node-ID in the SDWAN NLRI,
- the Tenant ID & Subnet ID are represented by the SDWAN VPN ID in the Client UPDATE.

The format of this Sub-TLV is as follows:

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   ID Length   |       Nonce Length            |I|   Flags     |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                              Rekey                            |
|                             Counter                           |
+---------------------------------------------------------------+
|                IPsec SA Identifier                            |
+---------------------------------------------------------------+
|                                                               |
~                           Nonce Data                          ~
|                                                               |
+---------------------------------------------------------------+
```

IPsec SA ID - The 4 bytes IPSec SA ID is to differentiate multiple IPsec SAs terminated at the edge. The IPsec SA ID can be used in the IPsec-SA-ID subTLV of a different BGP UPDATE message to refer to all the values carried in the IPsec Public Key SubTLV and the IPsec SA Proposal Sub-TLV that are in the same BGP UPDATE message as the IPsec SA Nonce sub-TLV.

7.2. IPsec Public Key Sub-TLV

The IPsec Public Key Sub-TLV is derived from the Key Exchange Sub-TLV described in [SECURE-EVPN] with an addition of Duration filed to define the IPSec SA life span. The edge nodes would pick the shortest duration value advertised by the peers.

The format of this Sub-TLV is as follows:

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   Diffie-Hellman Group Num    |           Reserved            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
~                       Key Exchange Data                       ~
|                                                               |
+---------------------------------------------------------------+
|                              Duration                         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

7.3. IPsec SA Proposal Sub-TLV

The IPsec SA Proposal Sub-TLV is to indicate the number of Transform Sub-TLVs. This Sub-TLV aligns with the sub-TLV structure from [SECURE-VPN].

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|    Transform Attr Length      |Transform Type |    Reserved.  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     Transform ID              |          Reserved             |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
~                        Transform Attributes                   ~
|                                                               |
+---------------------------------------------------------------+
```

The Transform Type and the Transform Attributes Sub-sub-TLV are taken from the section 3.3.2 and 3.3.5 of RFC7296, respectively.

7.4. Simplified IPsec SA sub-TLV

For a simple SDWAN network with edge nodes supporting only a few pre-defined encryption algorithms, a simple IPsec sub-TLV can be used to encode the pre-defined algorithms, as below:

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|IPsec-simType |IPsecSA Length                  | Flag          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Transform    | Mode          | AH algorithms |ESP algorithms |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|         ReKey Counter (SPI)                                   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| key1 length  |         Key 1                                  ~
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| key2 length  |         Key 2                                  ~
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| key-i length |         Nonce                                  ~
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|        Duration                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Where:

- o IPsec-SimType=70: indicate the simplified IPsec SA attributes.
- o IPsec-SA subTLV Length (2 Byte): 25 (or more)
- o Flags: 1 octet of flags. None are defined at this stage. Flags SHOULD be set to zero on transmission and MUST be ignored on receipt.
- o Transform (1 Byte):
  - Transform = 1 means AH,
  - Transform = 2 means ESP, or
  - Transform = 3 means AH+ESP.
- o IPsec Mode (1 byte):
  - Mode = 1 indicates that the Tunnel Mode is used
  - Mode = 2 indicates that the Transport mode is used.
- o AH algorithms (1 byte): AH authentication algorithms supported, which can be md5 | sha1 | sha2-256 | sha2-384 | sha2-512 | sm3. Each SDWAN edge node can have multiple authentication algorithms; send to its peers to negotiate the strongest one.
- o ESP algorithms (1 byte): ESP authentication algorithms supported, which can be md5 | sha1 | sha2-256 | sha2-384 | sha2-512 | sm3. Each SDWAN edge node can have multiple authentication algorithms; send to its peers to negotiate the strongest one. Default algorithm is AES-256.

   o When node supports multiple authentication algorithms, the initial UPDATE needs to include the "Transform Sub-TLV" described by [SECURE-EVPN] to describe all of the algorithms supported by the node.

- o Rekey Counter (Security Parameter Index)): 4 bytes
- o Public Key: IPsec public key
- o Nonce.IPsec Nonce
- o Duration: SA life span.

8. Error & Mismatch Handling

Each C-PE device advertises a SDWAN SAFI Underlay NLRI to the other C-PE devices via a BGP Route Reflector to establish pairwise SAs between itself and every other remote C-PEs. During the SAFI NLRI advertisement, the BGP originator would include either simple IPSec Security Association properties defined in IPSec SA Sub-TLV based on IPSec-SA-Type = 1 or full-set of IPSec Sub-TLVs including Nonce, Public Key, Proposal and number of Transform Sub-TLVs based on IPSec-SA-Type = 2.

The C-PE devices compare the IPSec SA attributes between the local and remote WAN ports. If there is a match on the SA Attributes between the two ports, the IPSec Tunnel is established.

The C-PE devices would not try to negotiate the base IPSec-SA parameters between the local and the remote ports in the case of simple IPSec SA exchange or the Transform sets between local and remote ports if there is a mismatch on the Transform sets in the case of full-set of IPSec SA Sub-TLVs.

As an example, using the Figure 1 in Section 3, to establish IPsec Tunnel between C-PE1 and C-PE2 WAN Ports A2 and B2 [A2: 192.10.0.10 <-> B2:192.0.0.1]:

C-PE1 needs to advertise the following attributes for establishing the IPsec SA:

```
  NH: 192.10.0.10
  SDWAN Node ID
  SDWAN-Site-ID
  Tunnel Encap Attr (Type=SDWAN)
       Transport-Sub-TLV for detailed information about the ISP3
       IPsec SA Nonce Sub-TLV,
```

```
          IPsec SA Public Key Sub-TLV,
          Proposal Sub-TLV with Num Transforms = 1
               {Transforms Sub-TLV - Trans 1}
```

C-PE2 needs to advertise the following attributes for establishing IPsec SA:

```
  NH: 192.0.0.1
  SDWAN Node ID
  SDWAN-Site-ID
  Tunnel Encap Attr (Type=SDWAN)
       Transport-Sub-TLV for the detailed information about the ISP1
       IPsec SA Nonce Sub-TLV,
       IPsec SA Public Key Sub-TLV,
       Proposal Sub-TLV with Num Transforms = 1
            {Transforms Sub-TLV - Trans 2}
```

As there is no matching transform between the WAN ports A2 and B2 in C-PE1 and C-PE2 respectively, there will be no IPsec Tunnel be established.

9. SDWAN BGP UPDATE Encoding Examples

9.1. Encoding example of WAN Port properties

The C-PE2 of the Figure 1 can send the following SDWAN UPDATE messages to advertise the properties associated with WAN Port 192.0.0.1 and WAN Port 170.0.0.1 respectively:

```
SDWAN NLRI: AFI=IPv4/IPv6 & SAFI = SDWAN;
         Color match with the Client route UPDATE's Color
         Extended Community
         local port id for WAN port 192.0.0.1
         Node-ID= 2.2.2.2 (C-PE2)
Tunnel-Type = Hybrid-SDWAN
Extended-Port-SubTLV for 192.0.0.1
```

```
   SDWAN NLRI: AFI=IPv4/IPv6 & SAFI = SDWAN;
            Color match with the Client route UPDATE's Color
            Extended Community
            local port id for WAN port 170.0.0.1
            Node-ID= 2.2.2.2 (C-PE2)
   Tunnel-Type = Hybrid-SDWAN
   Extended-Port-SubTLV for 170.0.0.1
```

9.2. Encoding example of IPsec SA terminated at the C-PE2

The C-PE2 of the Figure 1 can send the following SDWAN UPDATE messages to advertise node level IPsec SA:

```
   SDWAN NLRI: AFI=IPv4/IPv6 & SAFI = SDWAN;
            Color match with the Client route UPDATE's Color
            Extended Community
            Port-ID=0
            Node-ID= 2.2.2.2 (C-PE2)
   Tunnel-Type = Hybrid-SDWAN
   IPsec-SA-ID Sub-TLV or IPsec SA Property Sub-TLVs
```

9.3. Encoding example #1 of using IPsec-SA-ID Sub-TLV

This section provides an encoding example for the following scenario:

- There are four IPsec SAs terminated at the same node.
- Two of the IPsec SAs use GRE (value =2) as Inner Encapsulation within the IPsec Tunnel
- Two of the IPsec SA uses VxLAN (value = 8) as the Inner Encapsulation within its IPsec Tunnel.

Here is the encoding for the scenario:

```
    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
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   | Tunnel-Type =SDWAN-Hybrid     |       Length =                |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   ~                      GRE Sub-TLV                              ~
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   | subTLV-Type = IPsec-SA-ID     |       Length =                |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                     IPsec SA Identifier = 1                   |
   +---------------------------------------------------------------+
   |                     IPsec SA Identifier = 2                   |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   ~                      VxLAN Sub-TLV                            ~
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |subTLV-Type = IPsec-SA-ID      |        Length=                |
   +-------------------------------+-------------------------------+
   |                     IPsec SA Identifier = 3                   |
   +-------------------------------+-------------------------------+
   |                     IPsec SA Identifier = 4                   |
   +---------------------------------------------------------------+
```

The Length of the Tunnel-Type = SDDWAN-Hybrid is the sum of the following:

- Tunnel-end-point sub-TLV total length
- The GRE Sub-TLV total length,
- The IPsec-SA-ID Sub-TLV length,
- The VxLAN sub-TLV total length, and
- The IPsec-SA-ID Sub-TLV length.

10. Manageability Considerations

Unlike MPLS VPN whose PE nodes are all controlled by the network operators, SDWAN edge nodes can be installed anywhere, in shopping malls, in 3rd party Cloud DCs, etc.

It is very important to ensure that client routes advertisement from an SDWAN edge node are legitimate. The RR needs to drop all the BGP Update messages from an SDWAN edge nodes that have invalid Route Targets.

11. Security Considerations

The document describes the encoding for SDWAN edge nodes to advertise its properties to their peers to its RR, which propagates to the intended peers via untrusted networks.

The secure propagation is achieved by secure channels, such as TLS, SSL, or IPsec, between the SDWAN edge nodes and the local controller RR.

SDWAN edge nodes might not have secure channels with the RR. In this case, BGP connection has be established over IPsec or TLS.

12. IANA Considerations

12.1. Hybrid (SDWAN) Overlay SAFI

IANA has assigned SAFI = 74 as the Hybrid (SDWAN)SAFI.

12.2. Tunnel Encapsulation Attribute Type

IANA is requested to assign a type from the BGP Tunnel Encapsulation Attribute Tunnel Types as follows:

Value	Description	Reference
25	SDWAN-Hybrid	[this document]

12.3. Tunnel Encapsulation Attribute Sub-TLV Types

IANA is requested to assign the following sub-Types in the BGP Tunnel Encapsulation Attribute Sub-TLVs registry:

Value	Type Description	Reference
64	IPSEC-SA-ID Sub-TLV	[Section 6.3]
65	Extended Port Property Sub-TLV	[Section 6.4]
66	Underlay Transport Sub-TLV	[Section 6.5]
67	IPsec SA Nonce Sub-TLV	[Section 7.1]
68	IPsec Public Key Sub-TLV	[Section 7.2]
69	IPsec SA Proposal Sub-TLV	[Section 7.3]
70	Simplified IPsec SA sub-TLV	[Section 7.4]

13. References

13.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[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>.

[RFC4760] Bates, T., Chandra, R., Katz, D., and Y. Rekhter, "Multiprotocol Extensions for BGP-4", RFC 4760, DOI 10.17487/RFC4760, January 2007, <https://www.rfc-editor.org/info/rfc4760>.

[RFC7296] C. Kaufman, et al, "Internet Key Exchange Protocol Version 2 (IKEv2)", RFC7296, Oct. 2014.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

[RFC9012] Patel, K., Van de Velde, G., Sangli, S., and J. Scudder, "The BGP Tunnel Encapsulation Attribute", RFC 9012, DOI 10.17487/RFC9012, April 2021, <https://www.rfc-editor.org/info/rfc9012>.

13.2. Informative References

[RFC8192] S. Hares, et al, "Interface to Network Security Functions (I2NSF) Problem Statement and Use Cases", July 2017

[RFC5521] P. Mohapatra, E. Rosen, "The BGP Encapsulation Subsequent Address Family Identifier (SAFI) and the BGP Tunnel Encapsulation Attribute", April 2009.

[RFC9061] Marin-Lopez, R., Lopez-Millan, G., and F. Pereniguez-Garcia, "A YANG Data Model for IPsec Flow Protection Based on Software-Defined Networking (SDN)", RFC 9061, DOI 10.17487/RFC9061, July 2021, <https://www.rfc-editor.org/info/rfc9061>.

[CONTROLLER-IKE] D. Carrel, et al, "IPsec Key Exchange using a Controller", draft-carrel-ipsecme-controller-ike-01, work-in-progress.

[LISP-GEOLOC] D. Farinacci, "LISP Geo-Coordinate Use-Case", draft-farinacci-lisp-geo-09, April 2020.

[SDN-IPSEC] R. Lopez, G. Millan, "SDN-based IPsec Flow Protection", draft-ietf-i2nsf-sdn-ipsec-flow-protection-07, Aug 2019.

[SECURE-EVPN] A. Sajassi, et al, "Secure EVPN", draft-sajassi-bess-secure-evpn-05, Oct 2021.

[VPN-over-Internet] E. Rosen, "Provide Secure Layer L3VPNs over Public Infrastructure", draft-rosen-bess-secure-l3vpn-00, work-in-progress, July 2018

[DMVPN] Dynamic Multi-point VPN: https://www.cisco.com/c/en/us/products/security/dynamic-multipoint-vpn-dmvpn/index.html

[DSVPN] Dynamic Smart VPN: http://forum.huawei.com/enterprise/en/thread-390771-1-1.html

[ITU-T-X1036] ITU-T Recommendation X.1036, "Framework for creation, storage, distribution and enforcement of policies for network security", Nov 2007.

[Net2Cloud-Problem] L. Dunbar and A. Malis, "Dynamic Networks to Hybrid Cloud DCs Problem Statement", draft-ietf-rtgwg-net2cloud-problem-statement-12, March 7, 2022.

[Net2Cloud-gap] L. Dunbar, A. Malis, and C. Jacquenet, "Networks Connecting to Hybrid Cloud DCs: Gap Analysis", draft-ietf-rtgwg-net2cloud-gap-analysis-07, July, 2020.

[RFC9012] K. Patel, et al "The BGP Tunnel Encapsulation Attribute", RFC9012, April 2021.

14. Acknowledgments

Acknowledgements to Venkit Kasiviswanathan, Wang Haibo, Shunwan Zhuang, Hao Weiguo, and ShengCheng for implementation contribution; Many thanks to Yoav Nir, Graham Bartlett, Jim Guichard, John Scudder, and Donald Eastlake for their review and suggestions.

This document was prepared using 2-Word-v2.0.template.dot.

Authors' Addresses

Linda Dunbar
Futurewei
Email: ldunbar@futurewei.com

Sue Hares
Hickory Hill Consulting
Email: shares@ndzh.com

Robert Raszuk
Arrcus
Email: robert@raszuk.net

Kausik Majumdar
Microsoft
Email: kmajumdar@microsoft.com

Gyan Mishra
Verizon Inc.
Email: gyan.s.mishra@verizon.com

Contributors' Addresses

Shunwan Zhuang
Huawei
Email: zhuangshunwan@huawei.com

Donald Eastlake
Futurewei
Email: d3e3e3@gmail.com