InterAS Option B

This chapter explains the different InterAS option B configuration options. The available options are InterAS option B, InterAS option B (with RFC 3107), and InterAS option B lite. The InterAS option B (with RFC 3107) implementation ensures complete IGP isolation between the data centers and WAN. When BGP advertises a particular route to ASBR, it also distributes the label which is mapped to that route.

Information About InterAS

An autonomous system (AS) is a single network or group of networks that is controlled by a common system administration group and using a single, clearly defined protocol. In many cases, virtual private networks (VPNs) extend to different ASes in different geographical areas. Some VPNs must extend across multiple service providers; these VPNs are called overlapping VPNs. The connection between ASes must be seamless to the customer, regardless of the complexity or location of the VPNs.

InterAS and ASBR

Separate ASes from different service providers can communicate by exchanging information in the form of VPN IP addresses. The ASBRs use EBGP to exchange that information. The IBGP distributes the network layer information for IP prefixes throughout each VPN and each AS. The following protocols are used for sharing routing information:

- Within an AS, routing information is shared using IBGP.

- Between ASes, routing information is shared using EBGP. EBGP allows service providers to set up an interdomain routing system that guarantees loop-free exchange of routing information between separate ASes.

The primary function of EBGP is to exchange network reachability information between ASes, including information about the list of AS routes. The ASes use EBGP border edge routers to distribute the routes, which includes label-switching information. Each border edge router rewrites the next-hop and MPLS labels.

InterAS configuration supported in this MPLS VPN can include an interprovider VPN, which is MPLS VPNs that include two or more ASes, connected by separate border edge routers. The ASes exchange routes use EBGP, and no IBGP or routing information is exchanged between the ASes.

Exchanging VPN Routing Information

ASes exchange VPN routing information (routes and labels) to establish connections. To control connections between ASes, the PE routers and EBGP border edge routers maintain a label forwarding information base (LFIB). The LFIB manages the labels and routes that the PE routers and EBGP border edge routers receive during the exchange of VPN information.

The ASes use the following guidelines to exchange VPN routing information:

- Routing information includes:
  - The destination network.
  - The next-hop field associated with the distributing router.
  - A local MPLS label
- A route distinguisher (RD1) is part of a destination network address. It makes the VPN IP route globally unique in the VPN service provider environment.

The ASBRs are configured to change the next-hop when sending VPN NLRIs to the IBGP neighbors. Therefore, the ASBRs must allocate a new label when they forward the NLRI to the IBGP neighbors.

InterAS Options

Nexus 3600 series switches support the following InterAS options:

- **InterAS option A** - In an interAS option A network, autonomous system border router (ASBR) peers are connected by multiple subinterfaces with at least one interface VPN that spans the two ASes. These ASBRs associate each subinterface with a VPN routing and forwarding (VRF) instance and a BGP session to signal unlabeled IP prefixes. As a result, traffic between the back-to-back VRFs is IP. In this scenario, the VPNs are isolated from each other and, because the traffic is IP Quality of Service (QoS) mechanisms that operate on the IP traffic can be maintained. The downside of this configuration is that one BGP session is required for each subinterface (and at least one subinterface is required for each VPN), which causes scalability concerns as the network grows.
- **InterAS option B** - In an interAS option B network, ASBR ports are connected by one or more subinterfaces that are enabled to receive MPLS traffic. A Multiprotocol Border Gateway Router (MP-BGP) session distributes labeled VPN prefixes between the ASBRs. As a result, the traffic that flows between the ASBRs is labeled. The downside of this configuration is that, because the traffic is MPLS, QoS mechanisms that are applied only to IP traffic cannot be carried and the VRFs cannot be isolated. InterAS

option B provides better scalability than option A because it requires only one BGP session to exchange all VPN prefixes between the ASBRs. Also, this feature provides nonstop forwarding (NSF) and Graceful Restart. The ASBRs must be directly connected in this option.

Some functions of option B are noted below:

- You can have an IBGP VPNv4/v6 session between Nexus 3600 series switches within an AS and you can have an EBGP VPNv4/v6 session between data center edge routers and WAN routers.
- There is no requirement for a per VRF IBGP session between data center edge routers, like in the lite version.
- – LDP distributes IGP labels between ASBRs.
- **InterAS option B (with BGP-3107 or RFC 3107 implementation)**
- You can have an IBGP VPNv4/v6 implementation between Nexus 3600 platform switches within an AS and you can have an EBGP VPNv4/v6 session between data center edge routers and WAN routers.
- BGP-3107 enables BGP packets to carry label information without using LDP between ASBRs.
- The label mapping information for a particular route is piggybacked in the same BGP update message that is used to distribute the route itself.
- When BGP is used to distribute a particular route, it also distributes an MPLS label which is mapped to that route. Many ISPs prefer this method of configuration since it ensures complete IGP isolation between the data centers.
- **InterAS option B lite** – Support for the InterAS option B feature is restricted in the Cisco NX-OS 6.2(2) release. Details are noted in the Configuring InterAS Option B (lite version) section.

Guidelines and Limitations for Configuring InterAS Option B

The InterAS option B feature is not supported with BGP confederation AS. However, the Option B implementation is supported on Cisco Nexus 3600 platform switches.

Configuring the Switch for InterAS Option B

You enable certain features on the switch to run InterAS option B.

Before you begin

The install feature-set mpls command is available only in the default VDC, and you must enable it in default VDC.

Configure VRFs on the DC edge switches with following steps:

SUMMARY STEPS

1. **configure terminal**
2. **install feature-set mpls**

3. **feature mpls ldp**
4. **feature mpls l3vpn**
5. **feature bgp**
6. **vrf-context** *vrf-name*
7. **rd** *route-target-ext-community*
8. **address-family {ipv4 | ipv6} unicast**
9. **route-target {import | export}** *route-target-ext-community*
10. **copy running-config startup-config**

DETAILED STEPS

Procedure

	Command or Action	Purpose

Step 1	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**install feature-set mpls** **Example:** `switch(config)# install feature-set mpls`	Installs the MPLS feature set in the default VDC. **Note** You can only install and enable MPLS in the default VDC. Use the no form of this command to uninstall the MPLS feature set
Step 3	**feature mpls ldp** **Example:** `switch(config)# feature mpls ldp`	Enables the MPLS LDP feature on the device.. **Note** When the MPLS LDP feature is disabled on the device, no LDP commands are available.
Step 4	**feature mpls l3vpn** **Example:** `switch(config)# feature mpls l3vpn`	Enables the MPLS Layer 3 VPN feature.
Step 5	**feature bgp** **Example:** `switch(config)# feature bgp`	Enables the BGP feature.
Step 6	**vrf-context** *vrf-name* **Example:** `switch(config)# vrf context VPN1`	Defines the VPN routing instance by assigning a VRF name and enters VRF configuration mode. The vrf-name argument is any case-sensitive, alphanumeric string up to 32 characters.

	Command or Action	Purpose
Step 7	**rd** *route-target-ext-community* **Example:** `switch(config-vrf)# rd100:1`	Configures the route distinguisher. The route-distinguisher argument adds an 8-byte value to an IPv4 prefix to create a VPN IPv4 prefix.
Step 8	**address-family {ipv4 | ipv6} unicast** **Example:** `switch(config-vrf)# address-family ipv4 unicast`	Specifies the IPv4 or IPv6 address family type and enters address family configuration mode.
Step 9	**route-target {import | export}** *route-target-ext-community* **Example:** `switch(config-vrf-af-ip4)# route-target import 1:1`	Specifies a route-target extended community for a VRF as follows: • The import keyword imports routing information from the target VPN extended community. • The export keyword exports routing information to the target VPN extended community. • The route-target-ext-community argument adds the route-target extended community attributes to the VRF's list of import or export route-target extended communities.
Step 10	**copy running-config startup-config** **Example:** `switch(config-vrf-af-ip4)# copy running-config startup-config`	(Optional) Copies the running configuration to the startup configuration.

Configuring BGP for InterAS Option B

Configure DC Edge switches with IBGP & EBGP VPNv4/v6 with the following steps:

Before you begin

To configure BGP for InterAS option B, you need to enable this configuration on both the IBGP and EBGP sides. Refer to Figure 1 for reference.

SUMMARY STEPS

1. **configure terminal**
2. **router bgp** *as-number*
3. **neighbor** *ip-address*
4. **remote-as** *as-number*
5. **address-family {vpnv4 | vpnv6} unicast**
6. **send-community {both | extended}**
7. **retain route-target all**
8. **vrf** *vrf-name*

9. **address-family {ipv4 | ipv6} unicast**
10. **exit**
11. **copy running-config startup-config**

DETAILED STEPS

Procedure

	Command or Action	Purpose

Step 1	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**router bgp** *as-number* **Example:** `switch(config)# router bgp 100`	Enters the router BGP configuration mode and assigns an autonomous system (AS) number to the local BGP speaker device.
Step 3	**neighbor** *ip-address* **Example:** `switch(config-router)# neighbor 10.0.0.2`	Adds an entry to the BGP or multiprotocol BGP neighbor table, and enters router BGP neighbor configuration mode.
Step 4	**remote-as** *as-number* **Example:** `switch(config-router-neighbor)# remote-as 200`	The as-number argument specifies the autonomous system to which the neighbor belongs.
Step 5	**address-family {vpnv4 | vpnv6} unicast** **Example:** `switch(config-router-neighbor)# address-family vpnv4 unicast`	Enters address family configuration mode for configuring IP VPN sessions.
Step 6	**send-community {both | extended}** **Example:** `switch(config-router-neighbor-af)# send-community both`	Specifies that a communities attribute should be sent to both BGP neighbors.
Step 7	**retain route-target all** **Example:** `switch(config-router-neighbor-af)# retain route-target all`	(Optional). Retains VPNv4/v6 address configuration on the ASBR without VRF configuration. **Note** If you have a VRF configuration on the ASBR, this command is not required.
Step 8	**vrf** *vrf-name* **Example:** `switch(config-router-neighbor-af)# vrf VPN1`	Associates the BGP process with a VRF.

	Command or Action	Purpose
Step 9	**address-family {ipv4 | ipv6} unicast** **Example:** `switch(config-router-vrf)# address-family ipv4 unicast`	Specifies the IPv4 or IPv6 address family and enters address family configuration mode.
Step 10	**exit** **Example:** `switch(config-vrf-af)# exit`	Exits IPv4 address family.
Step 11	**copy running-config startup-config** **Example:** `switch(config-router-vrf)# copy running-config startup-config`	(Optional) Copies the running configuration to the startup configuration.

Configuring the Switch for InterAS Option B (with RFC 3107 implementation)

You enable certain features on the switch to run InterAS option B.

Before you begin

Configure VRFs on the DC edge switches with following steps:

SUMMARY STEPS

1. **configure terminal**
2. **install feature-set mpls**
3. **feature mpls ldp**
4. **feature mpls l3vpn**
5. **feature bgp**
6. **vrf-context** *vrf-name*
7. **rd** *route-distinguisher*
8. **address-family {ipv4 | ipv6} unicast**
9. **route-target {import | export}** *route-target-ext-community*
10. **copy running-config startup-config**

DETAILED STEPS

Procedure

	Command or Action	Purpose

Step 1	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**install feature-set mpls** **Example:** `switch(config)# install feature-set mpls`	Installs the MPLS feature set in the default VDC. **Note** You can only install and enable MPLS in the default VDC. Use the no form of this command to uninstall the MPLS feature set
Step 3	**feature mpls ldp** **Example:** `switch(config)# feature mpls ldp`	Enables the MPLS LDP feature on the device.. **Note** When the MPLS LDP feature is disabled on the device, no LDP commands are available.
Step 4	**feature mpls l3vpn** **Example:** `switch(config)# feature mpls l3vpn`	Enables the MPLS Layer 3 VPN feature.
Step 5	**feature bgp** **Example:** `switch(config)# feature bgp`	Enables the BGP feature.
Step 6	**vrf-context** *vrf-name* **Example:** `switch(config)# vrf context VPN1`	Defines the VPN routing instance by assigning a VRF name and enters VRF configuration mode. The vrf-name argument is any case-sensitive, alphanumeric string up to 32 characters.
Step 7	**rd** *route-distinguisher* **Example:** `switch(config-vrf)# rd100:1`	Configures the route distinguisher. The route-distinguisher argument adds an 8-byte value to an IPv4 prefix to create a VPN IPv4 prefix.
Step 8	**address-family {ipv4 | ipv6} unicast** **Example:** `switch(config-vrf)# address-family ipv4 unicast`	Specifies the IPv4 or IPv6 address family type and enters address family configuration mode.
Step 9	**route-target {import | export}** *route-target-ext-community* **Example:**	Specifies a route-target extended community for a VRF as follows: • The import keyword imports routing information from the target VPN extended community.

	Command or Action	Purpose
	`switch(config-vrf-af-ip4)# route-target import 1:1`	• The export keyword exports routing information to the target VPN extended community. • The route-target-ext-community argument adds the route-target extended community attributes to the VRF's list of import or export route-target extended communities.
Step 10	**copy running-config startup-config** **Example:** `switch(config-vrf-af-ip4)# copy running-config startup-config`	(Optional) Copies the running configuration to the startup configuration.

Configuring BGP for InterAS Option B (with RFC 3107 implementation)

Configure DC Edge switches with IBGP & EBGP VPNv4/v6 along with BGP labeled unicast family with following steps:

Before you begin

Ensure that you are in the correct VDC (or use the **switchto vdc** command).

SUMMARY STEPS

1. **configure terminal**
2. **router bgp** *as-number*
3. **address-family {vpnv4 | vpnv6} unicast**
4. **redistribute direct route-map** *tag*
5. **allocate-label all**
6. **exit**
7. **neighbor** *ip-address*
8. **remote-as** *as-number*
9. **address-family {ipv4|ipv6} labeled-unicast**
10. **retain route-target all**
11. **exit**
12. **neighbor** *ip-address*
13. **remote-as** *as-number*
14. **address-family {vpnv4|vpnv6} unicast**
15. **exit**
16. **address-family {vpnv4|vpnv6} unicast**
17. **Repeat the process with ASBR2**
18. **copy running-config startup-config**

DETAILED STEPS

Procedure

	Command or Action	Purpose

Step 1	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**router bgp** *as-number* **Example:** `switch(config)# router bgp 100`	Enters the router BGP configuration mode and assigns an autonomous system (AS) number to the local BGP speaker device.
Step 3	**address-family {vpnv4 | vpnv6} unicast** **Example:** `switch(config-router-neighbor)# address-family vpnv4 unicast`	Enters address family configuration mode for configuring IP VPN sessions.
Step 4	**redistribute direct route-map** *tag* **Example:** `switch(config-router-af)# redistribute direct route-map loopback`	Redistributes directly connected routes using the Border Gateway Protocol.
Step 5	**allocate-label all** **Example:** `switch(config-router-af)# allocate-label all`	Configures ASBRs with the BGP labeled unicast address family to advertise labels for the connected interface.
Step 6	**exit** **Example:** `switch(config-router-af)# exit`	Exits address family router configuration mode and enters router BGP configuration mode.
Step 7	**neighbor** *ip-address* **Example:** `switch(config-router)# neighbor 10.1.1.1`	Configures the BGP neighbour's IP address, and enters router BGP neighbour configuration mode.
Step 8	**remote-as** *as-number* **Example:** `switch(config-router-neighbor)# remote-as 100`	Specifies the BGP neighbour's AS number.
Step 9	**address-family {ipv4|ipv6} labeled-unicast** **Example:** `switch(config-router-neighbor)# address-family ipv4 labeled-unicast`	Configures the ASBR with the BGP labeled unicast address family to advertise labels for the connected interface. **Note** This is the command that implements RFC 3107.

	Command or Action	Purpose
Step 10	**retain route-target all** **Example:** `switch(config-router-neighbor-af)# retain route-target all`	(Optional). Retains VPNv4/v6 address configuration on the ASBR without VRF configuration. **Note** If you have a VRF configuration on the ASBR, this command is not required.
Step 11	**exit** **Example:** `Switch(config-router-neighbor-af)# exit`	Exits router BGP neighbour address family configuration mode and returns to router BGP configuration mode.
Step 12	**neighbor** *ip-address* **Example:** `switch(config-router)# neighbor 10.1.1.1`	Configures a loopback IP address, and enters router BGP neighbor configuration mode..
Step 13	**remote-as** *as-number* **Example:** `switch(config-router-neighbor)# remote-as 100`	Specifies the BGP neighbour's AS number.
Step 14	**address-family {vpnv4|vpnv6} unicast** **Example:** `switch(config-router-vrf)# address-family ipv4 unicast`	Configures the ASBR with the BGP VPNv4 unicast address family.
Step 15	**exit** **Example:** `switch(config-vrf-af)# exit`	Exits IPv4 address family.
Step 16	**address-family {vpnv4|vpnv6} unicast** **Example:** `switch(config-router-vrf)# address-family ipv4 unicast`	Configures the ASBR with the BGP VPNv4 unicast address family.
Step 17	**Repeat the process with ASBR2**	Configures ASBR2 with option B (RFC 3107) settings and implements complete IGP isolation between the two data centers DC1 and DC2.
Step 18	**copy running-config startup-config** **Example:** `switch(config-router-vrf)# copy running-config startup-config`	(Optional) Copies the running configuration to the startup configuration.

Creating an ACL to filter LDP connections between the ASBRs (RFC 3107 implementation)

SUMMARY STEPS

1. **configure terminal**
2. **ip access-list** *name*
3. **[** sequence-number **]deny tcp any any eq** *packet-length*
4. **[** sequence-number **] deny tcp any eq** *packet-length* **any**
5. **[** sequence-number **] deny udp any any eq** *packet-length*
6. **[** sequence-number **] deny udp any eq** *packet-length* **any**
7. **[** sequence-number **] permit ip any any**
8. **exit**
9. **interface** *type number*
10. **mpls ip**
11. **ip access-group** *name* **in**
12. **ip access-group** *name* **out**
13. **end**

DETAILED STEPS

Procedure

	Command or Action	Purpose

Step 1	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**ip access-list** *name* **Example:** `switch(config)# ip access-list LDP`	Creates an access list and enters ACL configuration mode.
Step 3	**[** sequence-number **]deny tcp any any eq** *packet-length* **Example:** `switch(config-acl)# 10 deny tcp any any eq 646`	Executes the ACL instruction as per the specified sequence.
Step 4	**[** sequence-number **] deny tcp any eq** *packet-length* **any** **Example:** `switch(config-acl)# 20 deny tcp any eq 646 any`	Executes the ACL instruction as per the specified sequence.

	Command or Action	Purpose
Step 5	**[** sequence-number **] deny udp any any eq** *packet-length* **Example:** `switch(config-acl)# 30 deny udp any any eq 646`	Executes the ACL instruction as per the specified sequence.
Step 6	**[** sequence-number **] deny udp any eq** *packet-length* **any** **Example:** `switch(config-acl)# 20 deny udp any eq 646 any`	Executes the ACL instruction as per the specified sequence.
Step 7	**[** sequence-number **] permit ip any any** **Example:** `switch(config-acl)# 50 permit ip any any`	Executes the ACL instruction as per the specified sequence.
Step 8	**exit** **Example:** `switch(config-acl)# exit`	Exits ACL configuration mode and enters global configuration mode.
Step 9	**interface** *type number* **Example:** `switch(config)# interface ethernet 2/20`	Enters interface configuration mode.
Step 10	**mpls ip** **Example:** `switch(config-if)# mpls ip`	Configures MPLS hop-by-hop forwarding on this interface.
Step 11	**ip access-group** *name* **in** **Example:** `switch(config-if)# ip access-group LDP in`	Specifies that the ACL (named LDP created in the earlier steps) be applied to inbound traffic on the interface.
Step 12	**ip access-group** *name* **out** **Example:** `switch(config-if)# ip access-group LDP out`	Specifies that the ACL (named LDP created in the earlier steps) be applied to the outbound traffic on the interface.
Step 13	**end** **Example:** `switch(config-if)# end`	Exits interface configuration mode and returns to the privileged EXEC mode

Configuring InterAS Option B (lite Version)

Guidelines and Limitations for Configuring InterAS Option B lite

- The aggregation switch supports only local VRFs, and Nexus devices within an autonomous system (AS) are connected through a VRF implementation.

- Routes learned from the IBGP peer are not sent to the EBGP peer and routes learned from an EBGP peer are not sent to IBGP VPNv4/VPNv6 peers.
- The interAS option B with MP-BGP on the EBGP side does not work with MP-BGP on the IBGP side. One interface goes to the core and one interface goes to the Layer 3 VPN.
- MP-BGP Layer 3 VPN does not work within an AS.

Configuring the Switch for InterAS Option B (lite version)

You enable certain features on the switch to run interAS option B.

Before you begin

The install feature-set mpls command is available only in the default VDC, and you must enable it in default VDC.

SUMMARY STEPS

1. **configure terminal**
2. **install feature-set mpls**
3. **feature mpls ldp**
4. **feature mpls l3vpn**
5. **feature bgp**
6. **vrf-context** *vrf-name*
7. **rd** *route-distinguisher*
8. **address-family {ipv4 | ipv6} unicast**
9. **route-target {import | export}** *route-target-ext-community*
10. **copy running-config startup-config**

DETAILED STEPS

Procedure

	Command or Action	Purpose

Step 1	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**install feature-set mpls** **Example:** `switch(config)# install feature-set mpls`	Installs the MPLS feature set in the default VDC. **Note** You can only install and enable MPLS in the default VDC. Use the no form of this command to uninstall the MPLS feature set.

	Command or Action	Purpose
Step 3	**feature mpls ldp** **Example:** `switch(config)# feature mpls ldp`	Enables the MPLS LDP feature on the device. When the MPLS LDP feature is disabled on the device, no LDP commands are available.
Step 4	**feature mpls l3vpn** **Example:** `switch(config)# feature mpls l3vpn`	Enables the MPLS Layer 3 VPN feature.
Step 5	**feature bgp** **Example:** `switch(config)# feature bgp`	Enables the BGP feature.
Step 6	**vrf-context** *vrf-name* **Example:** `switch(config)# vrf-context VPN1`	Defines the VPN routing instance by assigning a VRF name and enters VRF configuration mode. The vrf-name argument is any case-sensitive, alphanumeric string up to 32 characters.
Step 7	**rd** *route-distinguisher* **Example:** `switch(config-vrf)# rd 100:1`	Configures the route distinguisher. The route-distinguisher argument adds an 8-byte value to an IPv4 prefix to create a VPN IPv4 prefix.
Step 8	**address-family {ipv4 | ipv6} unicast** **Example:** `switch(config-vrf)# address-family ipv4 unicast`	Specifies the IPv4 or IPv6 address family type and enters address family configuration mode.
Step 9	**route-target {import | export}** *route-target-ext-community* **Example:** `switch(config-vrf-af-ip4)# route-target import 1:1`	Specifies a route-target extended community for a VRF as follows: **Note** • The **import** keyword imports routing information from the target VPN extended community. • The **export** keyword exports routing information to the target VPN extended community. • The *route-target-ext-community* argument adds the route-target extended community attributes to the VRF's list of import or export route-target extended communities.
Step 10	**copy running-config startup-config** **Example:** `switch(config-router-vrf)# copy running-config startup-config`	(Optional) Copies the running configuration to the startup configuration.

Configuring BGP for InterAS Option B (lite Version)

Configure EBGP VPNv4/v6 on the DC Edge switches using the following steps:

SUMMARY STEPS

1. **configure terminal**
2. **router bgp** *as-number*
3. **neighbor** *ip-address*
4. **remote-as** *as-number*
5. **address-family {vpnv4 | vpnv6} unicast**
6. **send-community {both | extended}**
7. **vrf** *vrf-name*
8. **address-family {ipv4 | ipv6} unicast**
9. **exit**
10. **copy running-config startup-config**

DETAILED STEPS

Procedure

	Command or Action	Purpose

Step 1	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**router bgp** *as-number* **Example:** `switch(config)# router bgp 100`	Enters the router BGP configuration mode and assigns an autonomous system (AS) number to the local BGP speaker device.
Step 3	**neighbor** *ip-address* **Example:** `switch(config-router)# neighbor 10.0.0.2`	Adds an entry to the BGP or multiprotocol BGP neighbor table, and enters router BGP neighbor configuration mode.
Step 4	**remote-as** *as-number* **Example:** `switch(config-router-neighbor)# remote-as 200`	The as-number argument specifies the autonomous system to which the neighbor belongs.
Step 5	**address-family {vpnv4 | vpnv6} unicast** **Example:** `switch(config-router-neighbor)# address-family vpnv4 unicast`	Enters address family configuration mode for configuring IP VPN sessions.
Step 6	**send-community {both | extended}** **Example:**	Specifies that a communities attribute should be sent to both BGP neighbors.

	Command or Action	Purpose
	`switch(config-router-neighbor-af)# send-community both`	
Step 7	**vrf** *vrf-name* **Example:** `switch(config-router-neighbor-af)# vrf VPN1`	Associates the BGP process with a VRF.
Step 8	**address-family {ipv4 | ipv6} unicast** **Example:** `switch(config-router-vrf)# address-family ipv4 unicast`	Specifies the IPv4 or IPv6 address family and enters address family configuration mode.
Step 9	**exit** **Example:** `switch(config-vrf-af)# exit`	Exits IPv4 address family.
Step 10	**copy running-config startup-config** **Example:** `switch(config-router-vrf)# copy running-config startup-config`	(Optional) Copies the running configuration to the startup configuration.

Verifying InterAS Option B Configuration

To verify InterAS option B configuration information, perform one of the following tasks:

Command	Purpose
show bgp { vpnv4 | vpnv6 } unicast [*ip-prefix/length* [**neighbors** *neighbor*]] {**vrf**{*vrf-name* | **all** } | **rd** *route-distinguisher* }	Displays VPN routes from the BGP table.
show bgp ipv6 unicast [**vrf***vrf-name*]	Displays information about BGP on a VRF for 6VPE.
show forwarding { ip | ipv6 } route vrf *vrf-name*	Displays the IP forwarding table that is associated with a VRF. Check that the loopback addresses of the local and remote CE routers are in the routing table of the PE routers.
show { ip | ipv6 } bgp[**vrf** *vrf-name*]	Displays information about BGP on a VRF..
show ip route[*ip-address* [mask]] [*protocol*] **vrf** *vrf-name*	Displays the current state of the routing table. Use the ip-address argument to verify that CE1 has a route to CE2. Verify the routes learned by CE1. Make sure that the route for CE2 is listed.

Command	Purpose
show {**ip** | **ipv6**} **routevrf** *vrf-name*	Displays the IP routing table that is associated with a VRF. Check that the loopback addresses of the local and remote CE routers are in the routing table of the PE routers.
show running-config bgp	Displays the running configuration for BGP.
show running-config vrf *vrf-name*	Displays the running configuration for VRFs.
show vrf *vrf-name* **interface** *if-type*	Verifies the route distinguisher (RD) and interface that are configured for the VRF.
trace *trace destination* **vrf** *vrf-name*	Discovers the routes that packets take when traveling to their destination. The trace command can help isolate a problem if two routers cannot communicate.

Configuration Examples for Configuring InterAS Option B

This example shows how to configure InterAS Option B

```
!--Configure VRFs on the DC edge switches --!

configure terminal
install feature-set mpls
feature mpls ldp
feature mpls l3vpn
feature bgp
vrf context VPN1
rd 100:1
address-family ipv4 unicast
route-target import 1:1
copy running-config startup-config

!--Configure DC Edge switches with IBGP & EBGP VPNv4/v6 --!

configure terminal
router bgp 100
neighbor 10.0.0.2
remote-as 200
address-family vpnv4 unicast
send-community both
retain route-target all
vrf VPN1
address-family ipv4 unicast
exit
copy running-config startup-config
```

This example shows how to configure InterAS Option B (RFC 3107)

```
!--Configure VRFs on the DC edge switches --!

configure terminal
install feature-set mpls
feature mpls ldp
```

```
feature mpls l3vpn
feature bgp
vrf context VPN1
rd 100:1
address-family ipv4 unicast
route-target import 1:1
copy running-config startup-config

!--Configure DC Edge switches with IBGP & EBGP VPNv4/v6 --!

configure terminal
router bgp 100
address-family ipv4 unicast
redistribute direct route-map loopback
allocate-label all
exit
neighbor 10.1.1.1
remote-as 100
address-family ipv4 labeled-unicast
retain route-target all
exit
neighbor 1.1.1.1
remote-as 100
address-family vpnv4 unicast
address-family vpnv6 unicast
!--Repeat the process with ASBR2. --!
copy running-config startup-config

!--Creating an ACL to filter LDP connection between the ASBRs (RFC 3107 implementation)--!

configure terminal
ip access-list LDP
10 deny tcp any any eq 646
20 deny tcp any eq 646 any
30 deny udp any any eq 646
40 deny udp any eq 646 any
50 permit ip any any
exit
interface ethernet 2/20
mpls ip
ip access-group LDP in
ip access-group LDP out
end
```