

Configuring Secure VXLAN EVPN Multi-Site Using CloudSec

This chapter contains the following sections:

- Secure VXLAN EVPN Multi-Sites, on page 1
- Guidelines and Limitations for Secure VXLAN EVPN Multi-Site Using CloudSec, on page 2
- Secure VXLAN EVPN Multi-Sites, on page 4
- Verify the Secure VXLAN EVPN Multi-Site Using CloudSec, on page 12
- Displaying Statistics for Secure VXLAN EVPN Multi-Site Using CloudSec, on page 18
- Configuration Examples for Secure VXLAN EVPN Multi-Site Using CloudSec, on page 19
- Migrating from Multi-Site with VIP to Multi-Site with PIP, on page 21
- Migration of Existing vPC BGWs, on page 21
- vPC Border Gateways, on page 22
- Enhanced Convergence, on page 23
- CloudSec Configurations, on page 24

Secure VXLAN EVPN Multi-Sites

Secure VXLAN EVPN Multi-Site using CloudSec ensures data security and data integrity for VXLAN-based Multi-Site fabrics. Using the cryptographic machinery of IEEE MACsec for UDP packets, this feature provides a secure tunnel between authorized VXLAN EVPN endpoints.

- The CloudSec session is point to point over DCI between border gateways (BGWs) on two different sites.
- All communication between sites uses Multi-Site PIP instead of VIP. For migration information, see Migrating from Multi-Site with VIP to Multi-Site with PIP, on page 21.
- Secure VXLAN EVPN Multi-Site using CloudSec is enabled on a per-peer basis. Peers that do not support CloudSec can operate with peers that do support CloudSec, but the traffic is unencrypted. We recommend allowing unencrypted traffic only during migration from non-CloudSec-enabled sites to CloudSec-enabled sites.
- CloudSec key exchange uses BGP while MACsec uses the MACsec Key Agreement (MKA). The CloudSec control plane uses the BGP IPv4 address family to exchange the key information. CloudSec keys are carried as part of Tunnel Encapsulation (tunnel type 18) attribute with BGP IPv4 routes using underlay BGP session.

Key Lifetimes and Hitless Key Rollovers

A CloudSec keychain can have multiple pre-shared keys (PSKs), each configured with a key ID and an optional lifetime. Pre-shared keys are seed keys used to derive further keys for traffic encryption and integrity validation. A list of pre-shared keys can be configured in a keychain with different lifetimes.

- A key lifetime specifies when the key expires. CloudSec rolls over to the next configured pre-shared key in the keychain after the lifetime expires.
- The time zone of the key can be local or UTC. The default time zone is UTC.
- In the absence of a lifetime configuration, the default lifetime is unlimited.

When the lifetime of the first key expires, it automatically rolls over to the next key in the list. If the same key is configured on both sides of the link at the same time, the key rollover is hitless. That is, the key rolls over without traffic interruption. The lifetime of the keys must be overlapped in order to achieve hitless key rollover.

To configure the CloudSec keychain, see Configure a CloudSec Keychain and Keys, on page 7.

Certificate Expirations

Certificates are used for exchanging Master Session Keys. When certificates expire, no further MSK rekeys will happen. The current secured sessions will continue to stay up and SAK rekeys will happen as configured. The certificate will have to be deleted from under the trustpoint and a new certificate needs to be imported for further MSK rekeys to occur.

Guidelines and Limitations for Secure VXLAN EVPN Multi-Site Using CloudSec

Follow these guidelines and limitations when deploying Secure VXLAN EVPN Multi-Site using CloudSec.

- Beginning with Cisco NX-OS Release 10.2(2)F, vPC Border Gateway is supported on Cisco Nexus 9300-FX2, -FX3 switches.
- Secure VXLAN EVPN Multi-Site using CloudSec is supported on Cisco Nexus 9300-FX2 platform switches beginning with Cisco NX-OS Release 9.3(5).
- Secure VXLAN EVPN Multi-Site using CloudSec (VXLAN Tunnel Encryption feature) is supported on Cisco Nexus 9300-FX3 platform switches from Cisco NX-OS Release 10.1(1) onwards.
- L3 interfaces and L3 port channels are supported as DCI links.
- When the DCI links are operational, CloudSec traffic that is destined for the switch must enter through the DCI uplinks.
- Secure VXLAN EVPN Multi-Site using CloudSec is supported for sites that are connected through a route server or sites that are connected using full mesh (without a route server). For sites that are connected through a route server, upgrade the server to Cisco NX-OS Release 9.3(5) or a later release and follow the instructions in Enable CloudSec VXLAN EVPN Tunnel Encryption, on page 4.

- VXLAN Tunnel Encryption feature is not supported on Cisco Nexus 9348GC-FX3, 9348GC-FX3PH, and 9332D-H2R, 93400LD-H1, 9364C-H1 switches.
- ICV is disabled by default in Cisco NX-OS Release 9.3(7). ICV should be disabled on the node when forming cloudsec tunnel sessions with node from the previous release (Cisco NX-OS Release 9.3(6)).
- Beginning with Cisco NX-OS Release 10.3.3, VXLAN Tunnel Encryption feature can be configured using Pre Shared Keys (PSK) or certificates using the Public Key Infrastructure(PKI).
- All of the BGWs on the same site should be configured for Secure VXLAN EVPN Multi-Site using CloudSec.
- Secure VXLAN EVPN Multi-Site using CloudSec on DCI links and MACsec on the internal fabric can coexist. However, they can't be enabled simultaneously on the same port or port group (MAC ID).
- Secure VXLAN EVPN Multi-Site using CloudSec peers must have the same keychain configuration in order to decrypt the secure traffic between them.
- A maximum of 60 peers are supported in the BGP IPv4 update of security key distribution in the Cisco Nexus 9300-FX2 family switches.
- Beginning with Cisco NX-OS Release 10.2(3), BGP IPv4 update of security key distribution is supported on Cisco Nexus 9300-FX3 platform switches.
- In order to keep a session alive when all keys with an active timer expire, configure no more than one key per keychain without a lifetime. As a best practice, we recommend configuring a lifetime for each key.
- CloudSec keys are exchanged between BGWs using Tunnel Encapsulation attribute with BGP IPv4 routes using underlay BGP session.

  If this attribute do not get propagated by intermediate nodes, you have to configure direct BGP IPv4 unicast session between the CloudSec end point nodes i.e., BGWs.
- Direct eBGP peering must be established between BGWs in each site if:
  - BGP is used as the IPv4 unicast routing protocol, but the Tunnel Encryption attribute is not propagated through DCI.
  - A routing protocol other than BGP is used for IPv4 unicast routing in the DCI (e.g., OSPF).
- eBGP peering is to be established over a Loopback interface that is different from the following interface:
  - The tunnel-encryption source-interface
  - The nve source-interface
- eBGP peering must filter the loopback IP used as the source of the adjacency. For example, if Loopback10 is used to establish eBGP peering for CloudSec, the IP of Lo10 should not be advertised over this adjacency.
- Secure VXLAN EVPN Multi-Site using CloudSec doesn't support the following:
  - Directly connected L2 hosts on Anycast Border Gateways
  - IP unnumbered configurations on the DCI interface
  - DCI multicast underlay

  - OAM pathtrace
  - TRM
  - VIP-only model on border gateways

- Beginning with Cisco NX-OS Release 10.5(3)F, Secure VXLAN EVPN Multi-Site using CloudSec supports downstream VNI feature on Cisco Nexus 9300-FX/FX2/FX3/GX/GX2/H2R/H1 Series switches, 9500 Series switches with X97160YC-EX and 9700-FX/GX/FX3 line cards.
- If CloudSec is enabled, non-disruptive ISSU is not supported.
- Different certificate types (SUDI, 3rd party RSA, 3rd party ECC) cannot be mixed in Cloudsec PKI deployments. All nodes should have the same type of certificates
- Nodes with different RSA key sizes are compatible for encryption/decryption.
- PSK and PKI sessions cannot coexist in deployments.
- Size of certificates should not exceed 1.5KB (2048 bit key size).
- MCT-less VPC BGWs is not supported.
- Migration between different certificate types can be done by moving to should-secure, removing trustpoint config from all participating nodes and then, configuring the new trustpoint on all nodes.
- When Cloudsec is initially enabled with the **feature tunnel-encryption** command, the vPC peer-link port-channel and its physical member interfaces will flap.
- Feature **dci-advertise-pip** is not supported on Cisco Nexus 9700-FX/FX3 line cards, and 9500 Series switches with FM-E/FM-E2 fabric modules.

Secure VXLAN EVPN Multi-Sites

Follow these procedures to configure Secure VXLAN EVPN Multi-Site using CloudSec:

Enable CloudSec VXLAN EVPN Tunnel Encryption

Before you begin

Configure BGP peers in the IPv4 unicast address family. Make sure that the IPv4 prefix is propagated with the tunnel community attribute that carries CloudSec keys.

Configure VXLAN EVPN Multi-Site and use the following commands to ensure that peer IP addresses are advertised for CloudSec VXLAN EVPN Tunnel Encryption:

```
evpn multisite border-gateway ms-id
dci-advertise-pip
```



Caution Configuring VXLAN EVPN Multi-Site without **dci-advertise-pip** reverts border gateways to VIP-only mode, which is not supported for CloudSec VXLAN EVPN Tunnel Encryption.

You have two options for sites that are connected through a route server:

- Keep dual RDs enabled – This default behavior ensures that the memory scale remains the same from previous releases in order to handle leaf devices with limited memory. All same-site BGWs use the same RD value for reoriginated routes while advertising EVPN routes to the remote BGW.
- Disable dual RDs – If you don't have memory limitations on leaf devices, you can configure the **no dual rd** command on the BGW. Different RD values are used for reoriginated routes on the same BGWs while advertising EVPN routes to the remote BGW.

Perform one of the following actions, depending on whether dual RDs are enabled on the BGW:

- If dual RDs are configured on the BGWs, follow these steps:
  1. Apply BGP additional paths on the BGW.

```
router bgp as-num
    address-family l2vpn evpn
      maximum-paths number
    additional-paths send
    additional-paths receive
```

  2. Configure multipath for each L3VNI VRF on the BGW.

```
vrf evpn-tenant-00001
  address-family ipv4 unicast
    maximum-paths 64
  address-family ipv6 unicast
    maximum-paths 64
```

  3. Apply BGP additional paths on the route server.

```
router bgp as-num
  address-family l2vpn evpn
    retain route-target all
    additional-paths send
    additional-paths receive
    additional-paths selection route-map name

route-map name permit 10
  set path-selection all advertise
```

- If **no dual rd** is configured on the BGWs or full mesh is configured, follow these steps:
  1. Configure the address family and maximum paths on the BGW.

```
router bgp as-num
  address-family l2vpn evpn
    maximum-paths number
```

  2. Configure multipath for each L3VNI VRF on the BGW.

```
vrf evpn-tenant-00001
    address-family ipv4 unicast
      maximum-paths 64
    address-family ipv6 unicast
      maximum-paths 64
```

Note BGP additional paths are not required on the route server.

Follow these steps to enable CloudSec VXLAN EVPN Tunnel Encryption.

SUMMARY STEPS

1. Use the **configure terminal** command to enter global configuration mode.
2. Use the [**no**] **feature tunnel-encryption** command to enable CloudSec VXLAN EVPN Tunnel Encryption.
3. Use the [**no**] **tunnel-encryption source-interface loopback** *number* command to specify the BGP loopback as the tunnel-encryption source interface. The IP address of the configured source interface is used as the prefix to announce CloudSec VXLAN EVPN Tunnel Encryption key routes.
4. Use the **tunnel-encryption icv** command to enable the Integrity Check Value (ICV).
5. (Optional) Use the **copy running-config startup-config** command to copy the running configuration to the startup configuration.

DETAILED STEPS

Procedure

Step 1 Use the **configure terminal** command to enter global configuration mode.

Example:

```
switch# configure terminal
switch(config)#
```

Step 2 Use the [**no**] **feature tunnel-encryption** command to enable CloudSec VXLAN EVPN Tunnel Encryption.

Example:

```
switch(config)# feature tunnel-encryption
```

Step 3 Use the [**no**] **tunnel-encryption source-interface loopback** *number* command to specify the BGP loopback as the tunnel-encryption source interface. The IP address of the configured source interface is used as the prefix to announce CloudSec VXLAN EVPN Tunnel Encryption key routes.

Example:

```
switch(config)# tunnel-encryption source-interface loopback 2
```

Step 4 Use the **tunnel-encryption icv** command to enable the Integrity Check Value (ICV).

Example:

```
switch(config)# tunnel-encryption icv
```

Step 5 (Optional) Use the **copy running-config startup-config** command to copy the running configuration to the startup configuration.

Example:

```
switch(config)# copy running-config startup-config
```

What to do next

After enabling CloudSec VXLAN EVPN tunnel encryption, you can follow any of the following procedure for authentication.

Configure a CloudSec Keychain and Keys, on page 7

or

Configuring CloudSec Certificate Based Authentication Using PKI, on page 8

Configure a CloudSec Keychain and Keys

Before you begin

Make sure that Secure VXLAN EVPN Multi-Site using CloudSec is enabled.

You can create a CloudSec keychain and keys on the device.

Procedure

Step 1 Use the **configure terminal** command to enter global configuration mode.

Example:

```
switch# configure terminal
switch(config)#
```

Step 2 Use the [**no**] **key chain** *name* **tunnel-encryption** command to create a CloudSec keychain and enter tunnel-encryption keychain configuration mode.

Example:

```
switch(config)# key chain kc1 tunnel-encryption
switch(config-tunnelencryptkeychain)#
```

Step 3 Use the [**no**] **key** *key-id* command to create a CloudSec key and enter tunnel-encryption key configuration mode.

Example:

```
switch(config-tunnelencryptkeychain)# key 2000
switch(config-tunnelencryptkeychain-tunnelencryptkey)#
```

The range is from 1 to 32 octets, and the maximum size is 64.

Note The key must consist of an even number of characters.

Step 4 Use the [**no**] **key-octet-string** *octet-string* **cryptographic-algorithm** {**AES_128_CMAC** | **AES_256_CMAC**} command to configure the octet string for the key.

Example:

```
switch(config-tunnelencryptkeychain-tunnelencryptkey)#
key-octet-string abcdef0123456789abcdef0123456789
abcdef0123456789abcdef0123456789
cryptographic-algorithm AES_256_CMAC
```

The *octet-string* argument can contain up to 64 hexadecimal characters. The octet key is encoded internally, so the key in clear text does not appear in the output of the **show running-config tunnel-encryption** command.

Step 5 Use the [**no**] **send-lifetime** *start-time* **duration** *duration* command to configure a send lifetime for the key.

Example:

```
switch(config-tunnelencryptkeychain-tunnelencryptkey)# send-lifetime 00:00:00 May 06 2020 duration
100000
```

By default, the device treats the start time as UTC.

The *start-time* argument is the time of day and date that the key becomes active. The *duration* argument is the length of the lifetime in seconds. The range is from 1800 seconds to 2147483646 seconds (approximately 68 years).

Step 6 (Optional) Use the **show key chain** *name* command to display the keychain configuration.

Example:

```
switch(config-tunnelencryptkeychain-tunnelencryptkey)# show key chain kc1
```

Step 7 (Optional) Use the **copy running-config startup-config** command to copy the running configuration to the startup configuration.

Example:

```
switch(config-tunnelencryptkeychain-tunnelencryptkey)# copy running-config startup-config
```

What to do next

Configure a CloudSec VXLAN EVPN Tunnel Encryption policy.

Configuring CloudSec Certificate Based Authentication Using PKI

This chapter contains the following sections:

Attach a Certificate to CloudSec

Before you begin

See Configuring PKI to know how to configure a trustpoint and install or import a valid certificate.

You may associate the Cisco NX-OS device with a trust point CA. Cisco NX-OS supports RSA algorithm and ECC (224 and 521 bit) algorithm certificates. Follow the below steps to associate trustpoint or Secure Unique Device Identifier (SUDI) to cloudsec. User need to execute any one of the following commands.

Procedure

Step 1 Use the **tunnel-encryption pki trustpoint** *name* command to associate a trustpoint to cloud security.

Example:

```
switch# tunnel-encryption pki trustpoint myCA_2K
switch(config)#
```

Dynamic change of trustpoint labels cannot be done because it will disrupt data traffic

Step 2 Use the **tunnel-encryption pki sudi** *name* command to associate SUDI to cloud security.

Example:

```
switch(config)# tunnel-encryption pki sudi
switch(config-trustpoint)#
```



Note Cisco devices have a unique identifier known as the Secure Unique Device Identifier (SUDI) Certificate. This hardware Certificate may be leveraged in lieu of Step 1.

Configure the Separate Loopback

Execute any one of the following steps to configure PKI loopback.

Procedure

Step 1 Use the **tunnel-encryption pki source-interface** *loopback* command to configure a separate loopback.

Example:

```
switch# tunnel-encryption pki source-interface loopback0
switch(config)#
```

Or execute the command in the next step.

Step 2 Use the **tunnel-encryption pki source-interface cloudsec-loopback** command to use the same loopback as the cloudsec source interface loopback.

Example:

```
switch(config)# tunnel-encryption pki source-interface cloudsec-loopback
```

Uses the same loopback as cloudsec source interface loopback.

Configure a CloudSec Policy

Before you begin

Make sure that Secure VXLAN EVPN Multi-Site using CloudSec is enabled.

You can create multiple CloudSec policies with different parameters. However, only one policy can be active on an interface.

Procedure

Step 1 Use the **configure terminal** command to enter global configuration mode.

Example:

```
switch# configure terminal
            switch(config)#
```

Step 2 (Optional) Use the [**no**] **tunnel-encryption must-secure-policy** command to ensure that no unencrypted packets are sent over the wire for the session. Packets that are not carrying CloudSec headers are dropped.

Example:

```
switch(config)# tunnel-encryption must-secure-policy
```

Step 3 Use the [**no**] **tunnel-encryption policy***name* command to create a CloudSec policy.

Example:

```
switch(config)# tunnel-encryption policy p1
            switch(config-tunenc-policy)#
```

Step 4 (Optional) Use the [**no**] **cipher-suite***name* command to configure one of the following ciphers: GCM-AES-XPN-128 or GCM-AES-XPN-256. The default value is GCM-AES-XPN-256.

Example:

```
switch(config-tunenc-policy)# cipher-suite GCM-AES-XPN-256
```

Step 5 (Optional) Use the [**no**] **window-size***number* command to configure the replay protection window. The interface will not accept any packet that is less than the configured window size. The range is from 134217728 to 1073741823 IP packets. The default value is 268435456.

Example:

```
switch(config-tunenc-policy)# window-size 134217728
```

Step 6 (Optional) Use the [**no**] **sak-rekey-time***time* command to configure the time in seconds to force an SAK rekey. The range is from 1800 to 2592000 seconds. There is not a default value. We recommend using the same rekey value for all the peers.

Example:

```
switch(config-tunenc-policy)# sak-rekey-time 1800
```

Step 7 (Optional) Use the **show tunnel-encryption policy** command to display the CloudSec policy configuration.

Example:

```
switch(config-tunenc-policy)# show tunnel-encryption policy
```

Step 8 (Optional) Use the **copy running-config startup-config** command to copy the running configuration to the startup configuration.

Example:

```
switch(config-tunenc-policy)# copy running-config startup-config
```

What to do next

Configure CloudSec VXLAN EVPN Tunnel Encryption peers.

Configuring CloudSec Peers

This chapter contains the following sections.

Configure the CloudSec Peers

Before you begin

Enable Secure VXLAN EVPN Multi-Site using CloudSec.

You can configure the CloudSec peers.

Procedure

Step 1 Use the **configure terminal** command to enter global configuration mode.

Example:

```
switch# configure terminal
switch(config)#
```

Step 2 Use the [**no**] **tunnel-encryption peer-ip** *peer-ip-address* command to specify the IP address of the NVE source interface on the peer.

Example:

```
switch(config)# tunnel-encryption peer-ip 33.1.33.33
```

Step 3 Use the [**no**] **keychain** *name* **policy** *name* command to attach a policy to a CloudSec peer. Step 4 is an alternative to this step.

Example:

```
switch(config)# keychain kc1 policy p1
```

Step 4 Use the **pki policy** *policy name* command to attach a CloudSec policy to a peer with PKI.

Example:

```
switch(config)# pki policy p1
```

What to do next

Configure CloudSec VXLAN EVPN Tunnel Encryption on an uplink interface.

Enable Secure VXLAN EVPN Multi-Site Using CloudSec on DCI Uplinks

Before you begin

Make sure that Secure VXLAN EVPN Multi-Site using CloudSec is enabled.

Follow these steps to enable Secure VXLAN EVPN Multi-Site using CloudSec on all DCI uplinks.

Note This configuration cannot be applied on Layer 2 ports.

Note When CloudSec is applied or removed from an operational DCI uplink, the link will flap. The flap may not be instantaneous as the link may remain down for several seconds.

Procedure

Step 1 Use the **configure terminal** command to enter global configuration mode.

Example:

```
switch# configure terminal
switch(config)#
```

Step 2 Use the [**no**] **interface ethernet** *port/slot* command to enter interface configuration mode.

Example:

```
switch(config)# interface ethernet 1/1
switch(config-if)#
```

Step 3 Use the [**no**] **tunnel-encryption** command to enable Secure VXLAN EVPN Multi-Site using CloudSec on the specified interface.

Example:

```
switch(config-if)# tunnel-encryption
```

Verify the Secure VXLAN EVPN Multi-Site Using CloudSec

Procedure

Step 1 Verify the global configuration and policy settings for CloudSec.

a) Use the **show tunnel-encryption info global** command to display global policy modes and active peer counts.

Example:

```
switch# show tunnel-encryption info global
Global Policy Mode: Must-Secure
 SCI list: 0000.0000.0001.0002 0000.0000.0001.0004
 No. of Active Peers        : 1
```

b) Use the **show tunnel-encryption policy** [*policy-name*] command to display cipher suites, window sizes, and rekey timers.

Example:

```
switch# show tunnel-encryption policy
Tunnel-Encryption Policy      Cipher            Window     SAK Rekey time
----------------------------  ----------------  ---------  --------------
cloudsec                      GCM-AES-XPN-256   134217728  1800
p1                            GCM-AES-XPN-256   1073741823
system-default-tunenc-policy  GCM-AES-XPN-256   268435456
```

Step 2 Verify the operational status of CloudSec sessions.

a) Use the **show tunnel-encryption session** [**peer-ip** *peer-ip-address*] [**detail**] command to check session security status between endpoints.

Example:

```
switch# show tunnel-encryption session
Tunnel-Encryption  Peer Policy  Keychain  RxStatus        TxStatus
-----------------  -----------  --------  -----------     ---------------
33.1.33.33         p1           kc1       Secure (AN: 0)  Secure (AN: 2)
33.2.33.33         p1           kc1       Secure (AN: 0)  Secure (AN: 2)
33.3.33.33         p1           kc1       Secure (AN: 0)  Secure (AN: 2)
44.1.44.44         p1           kc1       Secure (AN: 0)  Secure (AN: 0)
44.2.44.44         p1           kc1       Secure (AN: 0)  Secure (AN: 0)
```

b) Use the **show tunnel-encryption session** command to display session information specifically for PKI Certificate Trustpoints.

Example:

```
switch# sh tunnel-encryption session
Tunnel-Encryption Peer   Policy                                    Keychain
RxStatus         TxStatus
------------------------ ----------------------------------------
---------------------------------------- ----------------- ------------------
20.20.20.2               p1                                       PKI: myCA (RSA)
Secure (AN: 0)    Secure (AN: 0)
32.11.11.4               p1                                       PKI: myCA (RSA)
Secure (AN: 0)    Secure (AN: 0)
```

Step 3 Verify BGP and IP routing information for tunnel encryption.

a) Use the **show bgp ipv4 unicast** *ip-address* command to display tunnel encapsulation attributes for BGP routes.

Example:

```
switch# show bgp ipv4 unicast 199.199.199.199  Source-loopback configured on peer BGW for CloudSec
BGP routing table information for VRF default, address family IPv4 Unicast
BGP routing table entry for 199.199.199.199/32, version 109
Paths: (1 available, best #1)
Flags: (0x8008001a) (high32 0x000200) on xmit-list, is in urib, is best urib route, is in HW
Multipath: eBGP

  Advertised path-id 1
  Path type: external, path is valid, is best path, no labeled nexthop, in rib
  AS-Path: 1000 200 , path sourced external to AS
    89.89.89.89 (metric 0) from 89.89.89.89 (89.89.89.89)
      Origin IGP, MED not set, localpref 100, weight 0
      Tunnel Encapsulation attribute: Length 120

  Path-id 1 advertised to peers:
    2.2.2.2
```

The IP address used in the example represents the source-loopback configured on the peer Border Gateway (BGW) for CloudSec.

b) Use the **show bgp l2vpn evpn** *mac-address* command to display EVPN address family routing and Ethernet Segment Identifier (ESI) information.

Example:

```
switch(config)# show bgp l2vpn evpn 0012.0100.000a
BGP routing table information for VRF default, address family L2VPN EVPN
Route Distinguisher: 110.110.110.110:32876
BGP routing table entry for [2]:[0]:[0]:[48]:[0012.0100.000a]:[0]:[0.0.0.0]/216, version 13198
Paths: (1 available, best #1)
Flags: (0x000202) (high32 00000000) on xmit-list, is not in l2rib/evpn, is not in HW
Multipath: eBGP

  Advertised path-id 1
  Path type: external, path is valid, is best path, no labeled nexthop
             Imported to 1 destination(s)
             Imported paths list: l2-10109
  AS-Path: 1000 200 , path sourced external to AS
    10.10.10.10 (metric 0) from 89.89.89.89 (89.89.89.89)
      Origin IGP, MED not set, localpref 100, weight 0
      Received label 10109
      Extcommunity: RT:100:10109 ENCAP:8
      ESI: 0300.0000.0000.0200.0309

  Path-id 1 not advertised to any peer

Route Distinguisher: 199.199.199.199:32876
BGP routing table entry for [2]:[0]:[0]:[48]:[0012.0100.000a]:[0]:[0.0.0.0]/216, version 24823
Paths: (1 available, best #1)
Flags: (0x000202) (high32 00000000) on xmit-list, is not in l2rib/evpn, is not in HW
Multipath: eBGP

  Advertised path-id 1
  Path type: external, path is valid, is best path, no labeled nexthop
             Imported to 1 destination(s)
             Imported paths list: l2-10109
  AS-Path: 1000 200 , path sourced external to AS
    9.9.9.9 (metric 0) from 89.89.89.89 (89.89.89.89)
      Origin IGP, MED not set, localpref 100, weight 0
      Received label 10109
      Extcommunity: RT:100:10109 ENCAP:8
      ESI: 0300.0000.0000.0200.0309

  Path-id 1 not advertised to any peer
```

c) Use the **show ip route** *ip-address* **vrf** *vrf-name* command to verify VRF routes and associated tunnel IDs.

Example:

```
switch(config)# show ip route 205.205.205.9 vrf vrf903
IP Route Table for VRF "vrf903"
'*' denotes best ucast next-hop
'**' denotes best mcast next-hop
'[x/y]' denotes [preference/metric]
'%<string>' in via output denotes VRF <string>

205.205.205.9/32, ubest/mbest: 2/0
    *via 9.9.9.9%default, [20/0], 11:06:32, bgp-100, external, tag 1000, segid: 900003 tunnelid:
 0x9090909 encap: VXLAN
```

```
   *via 10.10.10.10%default, [20/0], 3d05h, bgp-100, external, tag 1000, segid: 900003 tunnelid:
0xa0a0a0a encap: VXLAN
```

Step 4 Verify Layer 2 EVPN route details and encapsulation flags.

a) Use the **show l2route evpn mac evi** *evi-id* **mac** *mac-address* **detail** command to display MAC route resolution and ESI details.

Example:

b) Use the **show l2route evpn mac evi** *evi-id* **mac** *mac-address* **detail** command to verify labeled nexthops and asymmetric VNI flags for remote sites.

Example:

```
switch# show l2route evpn mac evi 109 mac 0012.0100.000a detail

Flags -(Rmac):Router MAC (Stt):Static (L):Local (R):Remote (V):vPC link
(Dup):Duplicate (Spl):Split (Rcv):Recv (AD):Auto-Delete (D):Del Pending
(S):Stale (C):Clear, (Ps):Peer Sync (O):Re-Originated (Nho):NH-Override
(Pf):Permanently-Frozen, (Orp): Orphan

Topology    Mac Address    Prod   Flags         Seq No     Next-Hops

----------- -------------- ------ ------------- ---------- --------------------------------------
109         0012.0100.000a BGP    Rcv             0        5.5.5.5 (Label: 20000)(Flags:
 Asy)
                                                                                    6.6.6.6 (Label:

 10000)(Flags: Asy)
           Route Resolution Type: ESI
           Forwarding State: Resolved (PL)
           Resultant PL: 5.5.5.5 (Label: 20000)(Flags: Asy)
                         6.6.6.6 (Label: 10000)(Flags: Asy)
           Sent To: L2FM
         ESI : 0300.0000.0000.0200.0309
         Encap: 1
```

Step 5 Verify NVE interface status and security configuration details.

a) Use the **show nve interface** *interface-id* **detail** command to verify Multi-site BGW and DCI advertisement status.

Example:

```
switch(config)# show nve interface nve1 detail
Interface: nve1, State: Up, encapsulation: VXLAN
 VPC Capability: VPC-VIP-Only [not-notified]
 Local Router MAC: 700f.6a15.c791
 Host Learning Mode: Control-Plane
 Source-Interface: loopback0 (primary: 14.14.14.14, secondary: 0.0.0.0)
 Source Interface State: Up
 Virtual RMAC Advertisement: No
 NVE Flags:
 Interface Handle: 0x49000001
 Source Interface hold-down-time: 180
 Source Interface hold-up-time: 30
 Remaining hold-down time: 0 seconds
 Virtual Router MAC: N/A
 Virtual Router MAC Re-origination: 0200.2e2e.2e2e
 Interface state: nve-intf-add-complete
 Multisite delay-restore time: 180 seconds
 Multisite delay-restore time left: 0 seconds
 Multisite dci-advertise-pip configured: True
```

```
 Multisite bgw-if: loopback1 (ip: 46.46.46.46, admin: Up, oper: Up)
 Multisite bgw-if oper down reason:
```

b) Use the **show running-config rpm** command to verify key chain configurations and key masking status.

Example:

```
switch# show running-config rpm
!Command: show running-config rpm
!Running configuration last done at: Mon Jun 15 14:41:40 2020
!Time: Mon Jun 15 15:10:27 2020

version 9.3(5) Bios:version 05.40
key chain inter tunnel-encryption
  key 3301
    key-octet-string 7 075f79696a58405441412e2a577f0f077d6461003652302552040a0b76015a504e370c
7972700604755f0e22230c03254323277d2f5359741a6b5d3a5744315f2f cryptographic-algorithm AES_256_CMAC
key chain kc1 tunnel-encryption
  key 3537
    key-octet-string 7
072c746f172c3d274e33592e22727e7409106d00372532575803780077755621 3d4e0c7c00770576772
d08515e0804553124577f5a522e046d6a5f485c35425f59 cryptographic-algorithm AES_256_CMAC
    send-lifetime local 09:09:40 Apr 15 2020 duration 1800
  key 2001
    key-octet-string 7
075f79696a58405441412e2a577f0f077d6461003652302552040a0b76015a504e370c7972700604755
f0e22230c03254323277d2f5359741a6b5d3a5744315f2f cryptographic-algorithm AES_256_CMAC
  key 2065
    key-octet-string 7
0729791f6f5e3d213347292d517308730c156c7737223554270f787c07722a513e450a0a0703070c062
e0256210d0e204120510d29222a051f1e594c2135375359 cryptographic-algorithm AES_256_CMAC
  key 2129
    key-octet-string 7
075c796f6f2a4c2642302f5c56790e767063657a4b564f2156777c0a020228564a32780e0472007005530
c5e560f04204056577f2a222d056d1f5c4c533241525d cryptographic-algorithm AES_256_CMAC
  key 2193
    key-octet-string 7
07577014195b402336345a5f260f797d7d6264044b50415755047a7976755a574d350b7e720a0202715d7
a50530d715346205d0c2d525c001f6b5b385046365a29 cryptographic-algorithm AES_256_CMAC

switch# configure terminal
switch(config)# key-chain tunnelencrypt-psk no-show
switch(config)# show running-config rpm

!Command: show running-config rpm
!Running configuration last done at: Mon Jun 15 15:10:44 2020
!Time: Mon Jun 15 15:10:47 2020

version 9.3(5) Bios:version 05.40
key-chain tunnelencrypt-psk no-show
key chain inter tunnel-encryption
  key 3301
    key-octet-string 7 ****** cryptographic-algorithm AES_256_CMAC
key chain kc1 tunnel-encryption
  key 3537
    key-octet-string 7 ****** cryptographic-algorithm AES_256_CMAC
    send-lifetime local 09:09:40 Apr 15 2020 duration 1800
  key 2001
    key-octet-string 7 ****** cryptographic-algorithm AES_256_CMAC
  key 2065
    key-octet-string 7 ****** cryptographic-algorithm AES_256_CMAC
  key 2129
    key-octet-string 7 ****** cryptographic-algorithm AES_256_CMAC
```

```
  key 2193
    key-octet-string 7 ****** cryptographic-algorithm AES_256_CMAC
```

c) Use the **show running-config cert-enroll** and **show crypto ca certificates** *trustpoint-label* commands to verify PKI trustpoints and certificate validity.

Example:

```
switch# show running-config cert-enroll
!Command: show running-config cert-enroll
!Running configuration last done at: Fri Apr 21 10:53:30 2023
!Time: Fri Apr 21 12:07:31 2023

version 10.3(3) Bios:version 05.47
crypto key generate rsa label myRSA exportable modulus 1024
crypto key generate rsa label myKey exportable modulus 1024
crypto key generate rsa label tmpCA exportable modulus 2048
crypto key generate ecc label src15_ECC_key exportable modulus 224
crypto ca trustpoint src15_ECC_CA
     ecckeypair switch_ECC_key and so on
     revocation-check  crl
crypto ca trustpoint myRSA
     rsakeypair myRSA
     revocation-check  crl
crypto ca trustpoint tmpCA
     rsakeypair tmpCA
     revocation-check  crl
crypto ca trustpoint myCA
     rsakeypair myKey
     revocation-check  crl
```

Example:

```
switch(config)# show crypto ca certificates myCA
Trustpoint: myCA
certificate:
subject=CN = switch, serialNumber = FBO22411ABC
issuer=C = US, ST = CA, L = San Jose, O = Org, OU = EN, CN = PKI, emailAddress = abc@xyz.com
serial=2F24FCE6823FCBE5A8AC72C82D0E8E24EB327B0C
notBefore=Apr 19 19:43:48 2023 GMT
notAfter=Aug 31 19:43:48 2024 GMT
SHA1 Fingerprint=D0:F8:1E:32:6E:6D:44:21:6B:AE:92:69:69:AD:88:73:69:76:B9:18
purposes: sslserver sslclient

CA certificate 0:
subject=C = US, ST = CA, L = San Jose, O = Org, OU = EN, CN = PKI, emailAddress = abc@xyz.com
issuer=C = US, ST = CA, L = San Jose, O = Cisco, OU = EN, CN = PKI, emailAddress = ca@ca.com
serial=1142A22DDDE63A047DE0829413359362042CCC31
notBefore=Jul 12 13:25:59 2022 GMT
notAfter=Jul 12 13:25:59 2023 GMT
SHA1 Fingerprint=33:37:C6:D5:F1:B3:E1:79:D9:5A:71:30:FD:50:E4:28:7D:E1:2D:A3
purposes: sslserver sslclient
```

Displaying Statistics for Secure VXLAN EVPN Multi-Site Using CloudSec

This section describes how to display and clear Secure VXLAN EVPN Multi-Site statistics using CloudSec.

You can display or clear Secure VXLAN EVPN Multi-Site using CloudSec statistics using the following commands:

- **show tunnel-encryption statistics** [**peer-ip** *peer-ip-address*] — Displays statistics for Secure VXLAN EVPN Multi-Site using CloudSec.
- **clear tunnel-encryption statistics** [**peer-ip** *peer-ip-address*] — Clears statistics for Secure VXLAN EVPN Multi-Site using CloudSec.

```
switch# show tunnel-encryption statistics
Peer 16.16.16.16 SecY Statistics:

SAK Rx Statistics for AN [0]:
Unchecked Pkts: 0
Delayed Pkts: 0
Late Pkts: 0
OK Pkts: 8170598
Invalid Pkts: 0
Not Valid Pkts: 0
Not-Using-SA Pkts: 0
Unused-SA Pkts: 0
Decrypted In-Pkts: 8170598
Decrypted In-Octets: 4137958460 bytes
Validated In-Octets: 0 bytes

SAK Rx Statistics for AN [3]:
Unchecked Pkts: 0
Delayed Pkts: 0
Late Pkts: 0
OK Pkts: 0
Invalid Pkts: 0
Not Valid Pkts: 0
Not-Using-SA Pkts: 0
Unused-SA Pkts: 0
Decrypted In-Pkts: 0
Decrypted In-Octets: 0 bytes
Validated In-Octets: 0 bytes

SAK Tx Statistics for AN [0]:
Encrypted Protected Pkts: 30868929
Too Long Pkts: 0
Untagged Pkts: 0
Encrypted Protected Out-Octets: 15758962530 bytes
```

Note In tunnel encryption statistics, if you observe a traffic drop coinciding with an increase in late packets, it could be due to any of the following reasons:

- The packets are being discarded because they are received outside the replay window.
- The tunnel encryption peers are out of sync.
- There is a valid security risk.

In these situations, you should reset the peer session by removing and then reconfiguring the tunnel-encryption peer on the corresponding remote peer, in order to synchronize them again.

Configuration Examples for Secure VXLAN EVPN Multi-Site Using CloudSec

This section provides configuration examples for Secure VXLAN EVPN Multi-Site using CloudSec.

The following example shows how to configure Secure VXLAN EVPN Multi-Site using keychain:

```
key chain kc1 tunnel-encryption
key 2006
key-octet-string 7 075f79696a58405441412e2a577f0f077d6461003652302552040
a0b76015a504e370c7972700604755f0e22230c03254323277d2f5359741a6b5d3a5744315f2f
cryptographic-algorithm AES_256_CMAC

feature tunnel-encryption
tunnel-encryption source-interface loopback4
tunnel-encryption must-secure-policy

tunnel-encryption policy p1
  window-size 1073741823

tunnel-encryption peer-ip 11.1.11.11
  keychain kc1 policy p1
tunnel-encryption peer-ip 11.2.11.11
  keychain kc1 policy p1
tunnel-encryption peer-ip 44.1.44.44
  keychain kc1 policy p1
tunnel-encryption peer-ip 44.2.44.44
  keychain kc1 policy p1

interface Ethernet1/1
  tunnel-encryption

interface Ethernet1/7
  tunnel-encryption

interface Ethernet1/55
  tunnel-encryption

interface Ethernet1/59
  tunnel-encryption

evpn multisite border-gateway 111
dci-advertise-pip
```

```
router bgp 1000
router-id 12.12.12.12
no rd dual
address-family ipv4 unicast
  maximum-paths 10
address-family l2vpn evpn
  maximum-paths 10
vrf vxlan-900101
address-family ipv4 unicast
  maximum-paths 10
address-family ipv6 unicast
  maximum-paths 10

show tunnel-encryption session
Tunnel-Encryption Peer  Policy  Keychain  RxStatus        TxStatus
----------------------  ------  --------  --------------  --------------
11.1.11.11               p1     kc1       Secure (AN: 0)  Secure (AN: 2)
11.2.11.11               p1     kc1       Secure (AN: 0)  Secure (AN: 2)
44.1.44.44               p1     kc1       Secure (AN: 0)  Secure (AN: 2)
44.2.44.44               p1     kc1       Secure (AN: 0)  Secure (AN: 2)
```

The following example shows how to configure Certificate based Secure VXLAN EVPN Multi-site using Clousec Certificate based Authentication.

```
feature tunnel-encryption

tunnel-encryption must-secure-policy
tunnel-encryption pki trustpoint myCA
tunnel-encryption pki source-interface loopback3
tunnel-encryption source-interface loopback2
tunnel-encryption policy with-rekey
  sak-rekey-time 1800
tunnel-encryption peer-ip 7.7.7.7
  pki policy system-default-tunenc-policy


interface Ethernet1/20
  tunnel-encryption

interface Ethernet1/21
  tunnel-encryption

interface Ethernet1/25/1
  tunnel-encryption
```

When a vPC BGW learns its peer vPC BGW's primary IP address through BGP and advertises that route to its eBGP peers, configure an outbound route map to prefer the path through the BGW that owns the address. This configuration is required to prevent silent drops of broadcast, unknown-unicast, and multicast traffic between sites.

The following example shows how to configure outbound route-map to make BGW's path as the best path.

```
ip prefix-list pip_ip seq 5 permit 192.0.2.12/32
route-map pip_ip permit 5
  match ip address prefix-list pip_ip
  set as-path prepend 65001 65001
route-map pip_ip permit 10

router bgp 65001
  neighbor 198.51.100.10
    inherit peer EBGP-PEERS
    remote-as 65002
```

```
address-family ipv4 unicast
  route-map pip_ip out
```

Migrating from Multi-Site with VIP to Multi-Site with PIP

Follow these steps for a smooth migration from Multi-Site with VIP to Multi-Site with PIP. The migration needs to be done one site at a time. You can expect minimal traffic loss during the migration.

1. Upgrade all BGWs on all sites to Cisco NX-OS Release 9.3(5) or a later release.
2. Configure BGP maximum paths on all BGWs. Doing so is required for ESI-based MAC multipath and BGP to download all of the next-hops for EVPN Type-2 and Type-5 routes.
3. Pick one site at a time for the migration.
4. Shut down the same-site BGWs except for one BGW. You can use the NVE **shutdown** command to shut down the BGWs.
5. To avoid traffic loss, wait a few minutes before enabling Multi-Site with PIP on the active BGW. Doing so allows the same-site shutdown BGWs to withdraw EVPN routes so remote BGWs send traffic to only the active BGW.
6. Enable Multi-Site with PIP on the active BGW by configuring the **dci-advertise-pip** command.

   The Multi-Site with PIP-enabled BGW advertises the EVPN EAD-per-ES route for the virtual ESI.

   The Multi-Site with PIP-enabled BGW advertises EVPN Type-2 routes with a virtual ESI, using the PIP address as the next-hop and the PIP interface MAC as the RMAC (if applicable) towards the DCI. For Type-5 routes, no virtual ESI is carried. There is no change in the advertisement of EVPN Type-2 and Type-5 routes towards the fabric.

   The remote BGW performs ESI-based MAC multipathing as MAC routes are received with ESI.
7. Unshut the same-site BGWs one at a time and enable Multi-Site with PIP by entering the **dci-advertise-pip** command.

   The remote BGW performs ESI-based MAC multipathing for MAC routes as ESI is the same from all same-site BGWs.

   On the remote BGW, BGP selects paths as multipath and downloads all next-hops for EVPN Type-5 routes.

Migration of Existing vPC BGWs

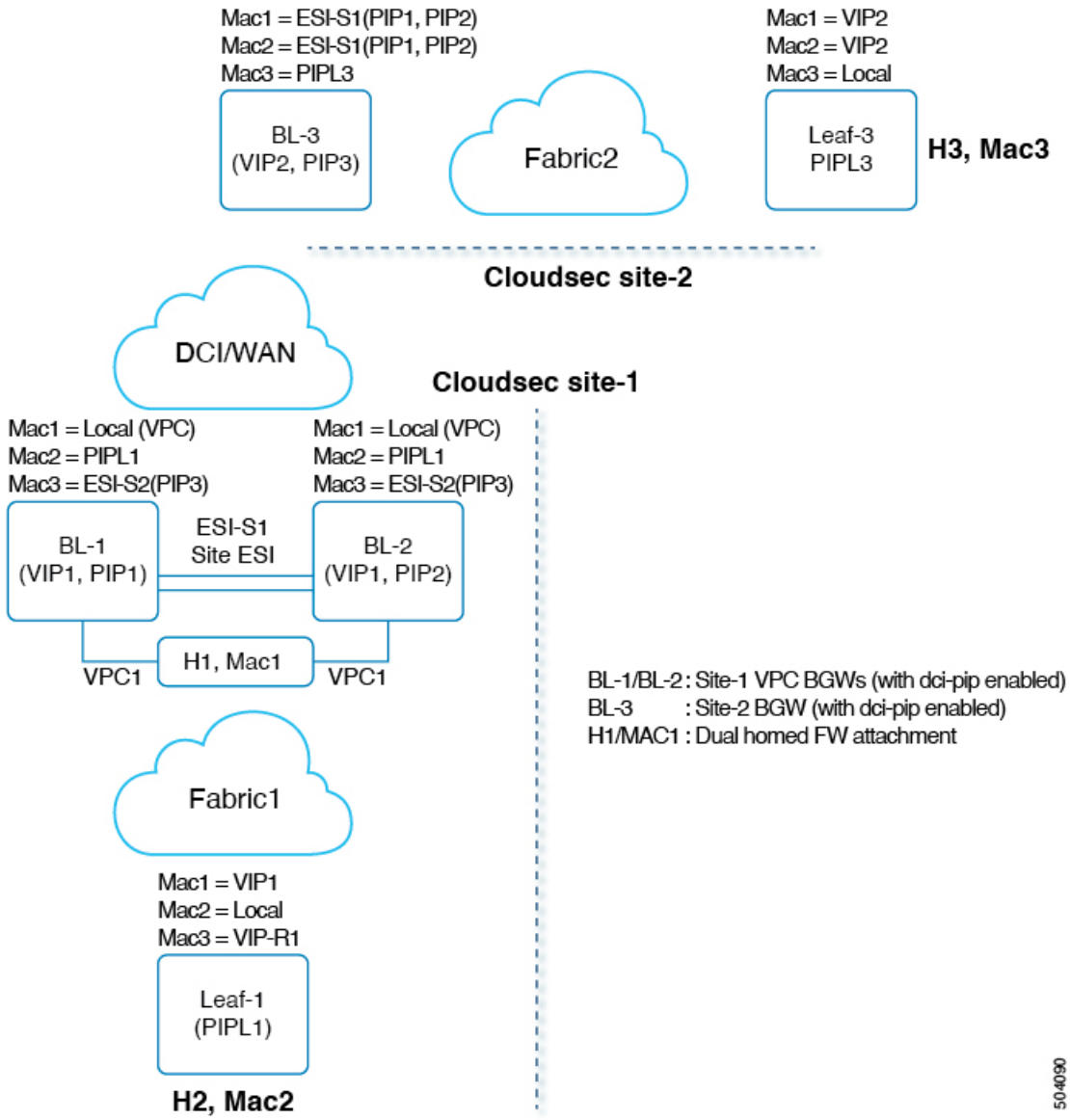
Follow these steps for a smooth migration of the existing vPC BGWs so that they can use Cloudsec. The migration needs to be done one site at a time. You can expect minimal traffic loss during the migration.

1. Upgrade both vPC BGWs to the latest image which has the vPC Cloudsec updates.
2. Shutdown interface nve1 on the vPC secondary.
3. Enable **dci-advertise-pip** on vPC primary.
4. With interface nve1 still in shut mode on vPC secondary, configure **dci-advertise-pip** on vPC secondary.
5. Unshut interface nve1 on vPC secondary.

vPC Border Gateways

The following topology illustrates the vPC Border Gateway (BGW) support for Cloudsec.

Figure 1: vPC BGW Support for Cloudsec



vPC is a dual-homed attachment/connection to the BGW. BGWs act virtually as a single VXLAN end point for redundancy and both switches function in active mode by sharing a common emulated/virtual ip-address (VIP). The VXLAN encapsulation over DCI is based on primary IP addresses of the BGW VTEPs.

In the above topology, Host H1/MAC1 is dually homed to Cloudsec enabled vPC BGWs BL-1/BL-2. H1 continues to be advertised with secondary loopback IP address of the vPC BGWs (VIP1) towards the fabric. However, towards the DCI, both BL-1/BL-2 advertise H1 with next-hop as PIP and site-ESI is also added to the Type-2 NLRI.

For Cloudsec feature on Anycast and vPC BGWs, dci-advertise-pip is configured to change the BGP procedures of how the Type-2/Type-5 routes are advertised to the DCI. All Type-2/Type-5 routes received from the site-internal network are advertised to the DCI with next-hop as PIP of the vPC BGW.

Both vPC BGWs advertise the routes with their primary IP address respectively. Site-ESI attribute is added to the Type-2 NLRIs. All dual attached hosts on the vPC BGWs are advertised with next-hop as PIP and site-ESI attribute is attached over DCI. All orphan hosts are advertised with next-hop as PIP towards DCI and the site-ESI attribute is not attached.

If vPC BGW learns peer vPC BGWs PIP address and advertises on DCI side, BGP path attributes from both vPC BGW will be same. Hence the DCI intermediate nodes may end up choosing the path from vPC BGW which does not own the PIP address. In this scenario MCT link is used for encrypted traffic coming from the remote site. The vPC BGW BGP the learns the peer vPC BGW's PIP address when:

- iBGP is configured between vPC BGWs.
- BGP is used as underlay routing protocol on fabric side.
- IGP used as underlay routing protocol, and IGP routes are redistributed into BGP.

When vPC BGW learns peer vPC BGW's PIP address in BGP, you need to configure the outbound route-map to make BGW's path as the best path.

On a remote site BGW, directly connected L3 host is learnt from both vPC BGWs. The path from directly connected BGW is usually preferred due to lower AS-path. If L3 host or L3 network is dually connected to vPC pair BGW, the local path is selected in both vPC pair.

Enhanced Convergence

Traditionally, a single loopback interface is configured as the NVE source interface, where both PIP and VIP of the vPC complex are configured. Beginning with Cisco NX-OS Release 10.3(2)F, you can configure a separate loopback for CloudSec-enabled vPC BGW. It is recommended to use separate loopback interfaces for source and anycast IP addresses under NVE for better convergence in vPC deployments. The IP address configured on the source-interface is the PIP of the vPC node, and the IP address configured on the anycast interface is the VIP of that vPC complex. Note that the secondary IP configured on the NVE source-interface will have no effect if the NVE anycast interface is also configured.

With separate loopbacks, the convergence for dual-attached EVPN Type-2 and Type-5 routes traffic destined for the DCI side will be improved.

Reference Information

Migration to Anycast Interface:

- If a user wants to specify an anycast interface, the user needs to unconfigure the existing source-interface and reconfigure with both source and anycast interfaces. This will lead to temporary traffic loss.
- For all green field deployments, it is recommended to configure both the source and anycast interface to avoid the convergence problem specified.

NVE Interface Configuration with Enhanced Convergence for vPC BGW CloudSec Deployments:

- The user needs to specify anycast interface along with NVE source-interface on vPC BGW.

- In today's VxLANv6 deployments, the provision to specify both source-interface and anycast interface is already present.
- In order to improve vPC convergence for VxLANv4, the anycast option is mandatory.

Configuration Example:

```
interface nve <number>
        source-interface <interface> [anycast <anycast-intf>]
```

iBGP Session Requirement:

- Underlay IPv4/IPv6 unicast iBGP session must be configured between vPC BGW peer nodes. This is to accommodate key propagation during the DCI isolation on any vPC BGW.

CloudSec Configurations

During migration to Auto keying, it is expected to send or receive clear traffic on a VTEP-to-VTEP session while the sites are still migrating to new configuration or functionalities. During this time, policy should be configured as **should-secure** to make sure unencrypted traffic is not dropped for the session.

1. Change tunnel-encryption config to **should-secure** on all nodes.
2. Perform migration one node at a time.
3. Remove the keychain and cloudsec policy from peer.
4. Configure trust point and certificate using a valid CA if using SSL certificates OR configure for SUDI certificates.
5. Attach the trust point to Cloudsec.
6. Apply the cloudsec policy back to the peer.
7. After all the nodes have been changed to autokeying, change the configuration to **must-secure** if needed.