Nexus 9000でのVXLAN VRFリークの設定と確認

内容

はじめに

このドキュメントでは、VXLAN環境でのVRFリークの設定および確認方法について説明します。

背景説明

VXLAN(Virtual Extensible LAN)環境では、VXLANホストをファブリックから外部ホストに接続するために、VRFリークとボーダーリーフデバイスを使用する必要があります。

VRFリークは、ネットワークのセグメント化とセキュリティを維持しながら、VXLANホストと外部ホスト間の通信を可能にするために不可欠です。

ボーダーリーフデバイスは、VXLANファブリックと外部ネットワークの間のゲートウェイとして機能し、この通信を促進する重要な役割を果たします。

このシナリオにおけるVRFリークの重要性は、次の文でまとめることができます。

1. 外部ネットワークとの相互接続:VRFリークにより、ファブリック内のVXLANホストがファブリック外の外部ホストと通信できます。これにより、外部ネットワーク（インターネットや他のデータセンターなど）でホストされるリソース、サービス、アプリケーションへのアクセスが可能になります。

2. ネットワークのセグメント化と分離:VRFリークにより、VXLANファブリック内でネットワークのセグメント化と分離が維持され、外部ネットワークとの選択的な通信が可能になります。これにより、VXLANホストはVRF割り当てに基づいて互いに分離された状態を維持しながら、必要に応じて外部リソースにアクセスできます。

3. ポリシーの適用:VRFリークにより、管理者はネットワークポリシーを適用し、VXLANホストと外部ホスト間のトラフィックフローを制御できます。これにより、通信で事前に定義されたセキュリティポリシーが使用され、機密リソースへの不正アクセスが防止されます。

4. 拡張性と柔軟性:VRFリークは、VXLANホストが外部ホストとシームレスに通信できるようにすることで、VXLAN導入の拡張性と柔軟性を強化します。VXLANと外部ネットワーク間でリソースの動的な割り当てと共有を可能にし、既存の設定を中断することなく、変化するネットワーク要件に適応します。

VRF(Virtual Routing and Forwarding)漏出でのルートのフィルタリングは、ネットワークセキュリティの維持、ルーティング効率の最適化、意図しないデータ漏出の防止のために非常に重要です。VRFリークにより、仮想ネットワーク間の通信が可能になると同時に、論理ネットワークとの分離が維持されます。

VRFリークにおけるルートのフィルタリングの重要性は、次の文でまとめることができます。

1. セキュリティ：ルートをフィルタリングすることにより、特定のルートだけがVRFインスタンス間でリークされ、不正アクセスやデータ漏洩のリスクが軽減されます。VRF境界の通過を許可するルートを制御することで、管理者はセキュリティポリシーを適用し、機密情報が

不正なエンティティにさらされるのを防ぐことができます。

2. 分離:VRFは、ネットワークのセグメント化と分離を実現するように設計されており、異なるテナントや部門が同じ物理インフラストラクチャ内で独立して運用できます。VRFリークでのルートのフィルタリングは、VRFインスタンス間でのルート伝搬の範囲を制限し、意図しない通信や潜在的なセキュリティの脆弱性を防ぐことで、この分離の維持に役立ちます。

3. ルーティングの最適化：ルートのフィルタリングにより、管理者はVRF間で必要なルートだけを選択的にリークし、ルーティング効率を最適化して、ネットワーク上の不要なトラフィックを削減できます。無関係なルートをフィルタリングすることで、管理者はトラフィックが最も効率的なパスを使用していることを確認しながら、輻輳と遅延を最小限に抑えることができます。

4. リソース使用率：ルートをフィルタリングすることにより、管理者はVRFインスタンス間のトラフィックフローを制御し、リソース使用率と帯域幅割り当てを最適化できます。これにより、ネットワークの輻輳を防止し、優先度の高いアプリケーションやサービスに重要なリソースを確実に割り当てることができます。

5. コンプライアンス:VRF漏出でのルートのフィルタリングにより、組織は規制要件や業界標準へのコンプライアンスを維持できます。許可されたエンティティだけにルートの漏洩を制限することで、組織はデータ保護規制に準拠していることを実証し、機密情報の整合性を確保できます。

6. きめ細かい制御：ルートのフィルタリングにより、管理者はVRFインスタンス間の通信をきめ細かく制御でき、固有の要件に基づいて特定のポリシーを定義できます。この柔軟性により、組織はさまざまなアプリケーション、ユーザ、または部門のニーズに合わせてネットワーク設定を調整できます。

前提条件

境界ルータを持つ既存のVXLAN環境

要件

次の項目に関する知識があることが推奨されます。

- NXOSプラットフォーム
- VXLAN
- VRF
- BGP

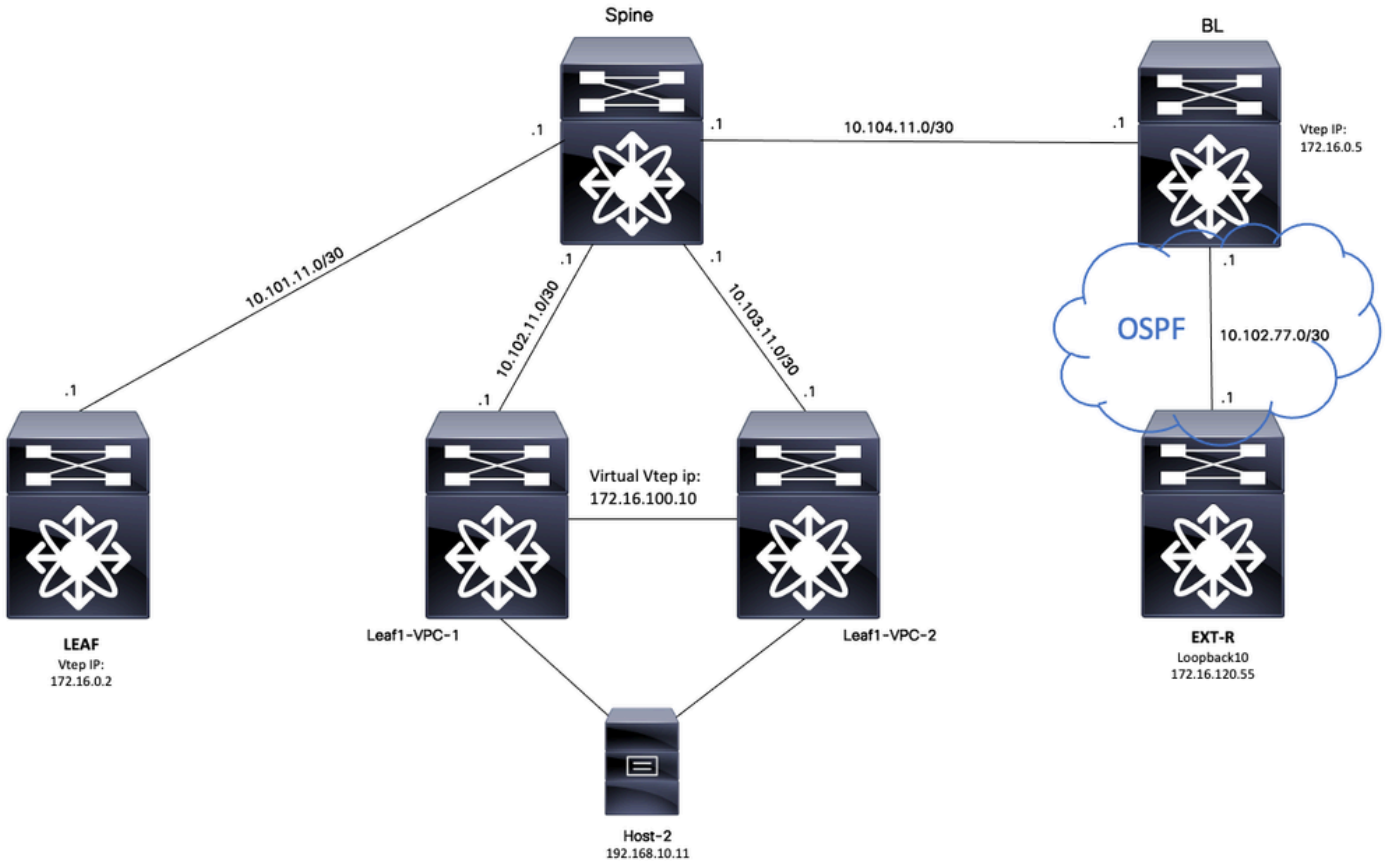
使用するコンポーネント

[名前(Name)]	Platform	バージョン
HOST-2	N9K-C92160YC-X	9.3(6)
リーフVPC-1	N9K-C93180YC-EX	9.3(9)

リーフVPC-2	N9K-C93108TC-EX	9.3(9)
リーフ	N9K-C9332D-GX2B	10.2(6)
ブルー	N9K-C9348D-GX2A	10.2(5)
EXT-R	N9K-C9348D-GX2A	10.2(3)
スパイン	N9K-C93108TC-FX3P	10.1(1)

"このドキュメントの情報は、特定のラボ環境にあるデバイスに基づいて作成されました。このドキュメントで使用するすべてのデバイスは、クリアな（デフォルト）設定で作業を開始しています。稼働中のネットワークで作業を行う場合、コマンドの影響について十分に理解したうえで作業してください」

図



アプリケーションとしてBGPを考慮すると、BGPはVRF間のリークを実行するために使用されるアプリケーションです

デフォルトのVRFからテナント：VRF

この例では、Border VTEP(BL)は、テナントVRFにリークされるデフォルトVRFのOSFPを介して外部デバイスから172.16.120.55を受信しています。

ルーティングテーブルの確認

```
BL# sh ip route 172.16.120.55
IP Route Table for VRF "default"
'*' denotes best ucast next-hop
'**' denotes best mcast next-hop
'[x/y]' denotes [preference/metric]
'%<string>' in via output denotes VRF <string>

172.16.120.55/32, ubest/mbest: 1/0
*via 10.105.100.2, Eth1/41.2, [110/2], 00:00:10, ospf-1, intra
```

フィルタルート

NXOSでは、ルートマップはルートをフィルタリングおよび再配布するためのパラメータとして必要です。この例では、プレフィックス172.16.120.55/32がフィルタリングされます。

設定

	コマンドまたはアクション	目的
手順 1	BL#端末の設定 Enter configuration commands, one per line.CNTL/Zで終了します。	コンフィギュレーションモードを開始します。
手順 2	BL(config)# ip prefix-list VXLAN-VRF-default-to-Tenant permit 172.16.120.55/32	プレフィックスリスト照合ホストを作成します。
手順 3	BL(config)# route-map VXLAN-VRF-default-to-Tenant	ルートマップを作成します。
手順 4	BL(config-route-map)# match ip address prefix-list VXLAN-VRF-default-to-Tenant	ステップ2で作成したmatch prefix-list。

BGPへのルートのインポート

ルートがデフォルトVRFに存在することが確認されたら、ルートをBGPプロセスにインポートする必要があります。

設定

	コマンドまたはアクション	目的

手順 1	BL#端末の設定 Enter configuration commands, one per line.CNTL/Z で終了します。	コンフィギュレーションモードを開始します。
手順 2	BL(config)# router bgp 65000	BGP設定に入ります。
手順 3	BL(config-router)# address-family ipv4 unicast	BGP address-famili IPV4と入力します。
手順 4	BL(config-router-af)# redistribute ospf 1 route-map VXLAN-VRF-default-to-Tenant	ステップ3で作成したルートマップを使用して、OSPFからBGPにルートを再配布します。

BGPテーブルの確認

```
BL(config-router-af)# show ip bgp 172.16.120.55
BGP routing table information for VRF default, address family IPv4 Unicast
BGP routing table entry for 172.16.120.55/32, version 16
Paths: (1 available, best #1)
Flags: (0x000002) (high32 00000000) on xmit-list, is not in urib

Advertised path-id 1
Path type: redist, path is valid, is best path, no labeled nexthop
AS-Path: NONE, path locally originated
0.0.0.0 (metric 0) from 0.0.0.0 (172.16.0.5)
Origin incomplete, MED 2, localpref 100, weight 32768
Extcommunity: OSPF RT:0.0.0.0:0:0
```

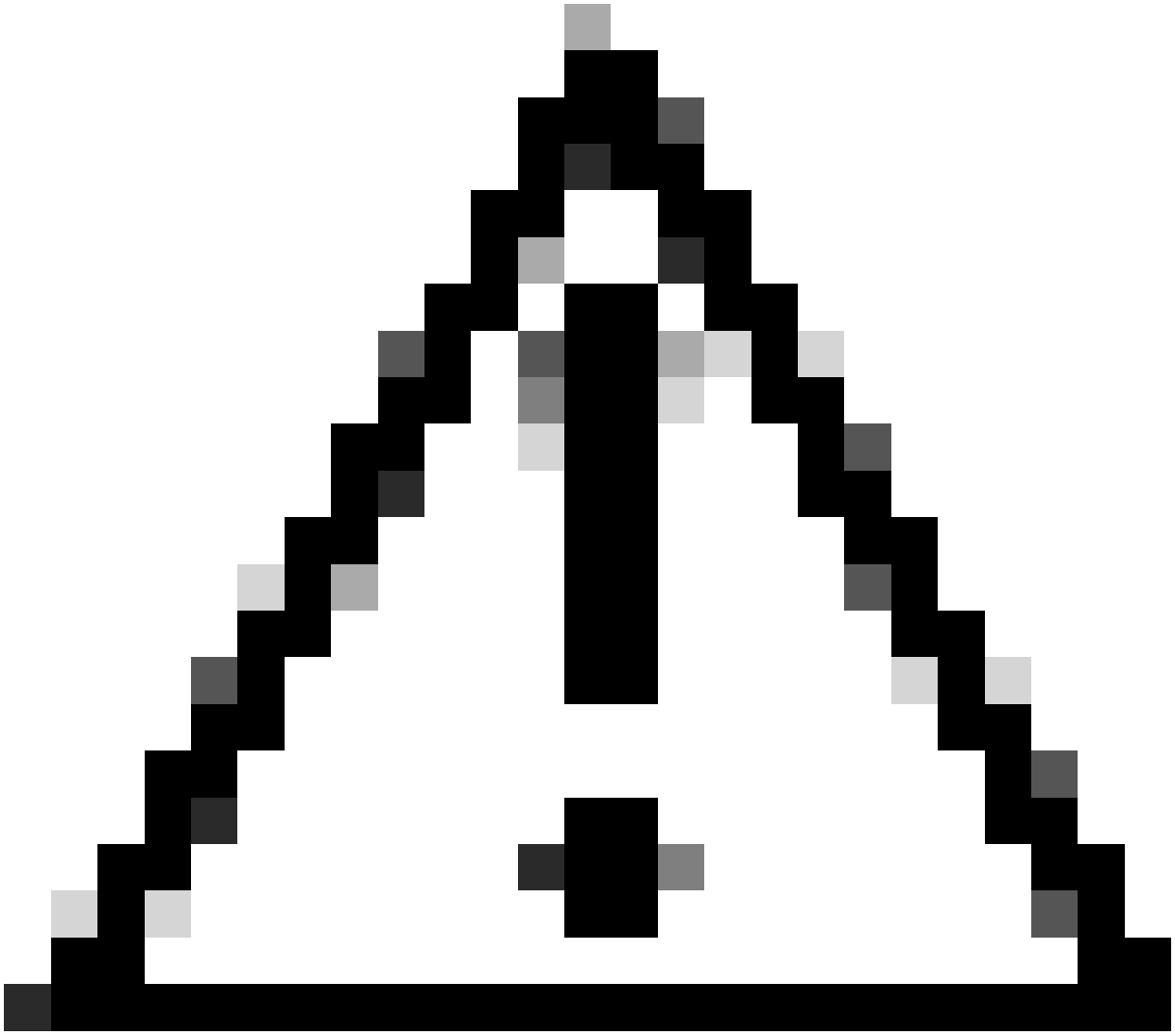
テナントVRFへのルートのインポート

ルートがBGPにインポートされると、ルートをターゲットVRF（テナントa）にインポートできるようになります。

設定

	コマンドまたはアクション	目的
手順 1	BL(config)# vrf context tenant-a	VRF設定を開始します。

手順2	BL(config-vrf)# address-family ipv4 unicast	IPV4アドレスファミリを開始します。
手順3	BL(config-vrf-af-ipv4)# import vrf default map VXLAN-VRF-default-to-Tenant advertise-vpn	VRFデフォルトからテナントVRFアドバタイジングVPNへのルートのインポート



注意：デフォルトでは、デフォルトのVRFからデフォルト以外のVRFにインポートできるIPプレフィックスの最大数は1000ルートです。この値は、VRF address-family IPV4:

import vrf <number of prefixes> default map <route-map name> advertise-vpnの下のコマンドで変更できます。

手順の概要

1. configure terminal
2. ip prefix-list VXLAN-VRF-default-to-Tenant permit 172.16.120.55/32を設定する
3. ルートマップVXLAN-VRF-default-to-Tenant
4. match ip address prefix-list VXLAN-VRF-default-to-Tenant
5. router bgp 65000
6. address-family ipv4 unicast
7. redistribute ospf 1 route-map VXLAN-VRF-default-to-Tenant
8. vrfコンテキストテナントa
9. address-family ipv4 unicast
10. import vrf default map VXLAN-VRF-default-to-Tenant advertise-vpn

確認

ルートがL2VPNにインポートされていることを確認します。

```
BL# sh bgp l2vpn evpn 172.16.120.55
BGP routing table information for VRF default, address family L2VPN EVPN
Route Distinguisher: 172.16.0.5:3 (L3VNI 303030)
BGP routing table entry for [5]:[0]:[0]:[32]:[172.16.120.55]/224, version 38
Paths: (1 available, best #1)
Flags: (0x000002) (high32 00000000) on xmit-list, is not in l2rib/evpn
Multipath: Mixed

Advertised path-id 1
Path type: local, path is valid, is best path, no labeled nexthop
Gateway IP: 0.0.0.0
AS-Path: NONE, path locally originated
172.16.0.5 (metric 0) from 0.0.0.0 (172.16.0.5)
Origin incomplete, MED 2, localpref 100, weight 32768
Received label 303030
Extcommunity: RT:65000:303030 ENCAP:8 Router MAC:20cf.ae54.fa3b
OSPF RT:0.0.0.0:0:0

Path-id 1 advertised to peers:
10.104.11.1
```

テナントVRFへのルートのインポートの確認

```
BL# sh ip route 172.16.120.55 vrf tenant-a
IP Route Table for VRF "tenant-a"
'*' denotes best ucast next-hop
```

```
'**' denotes best mcast next-hop
'[x/y]' denotes [preference/metric]
'%<string>' in via output denotes VRF <string>

172.16.120.55/32, ubest/mbest: 1/0
*via 172.16.0.5%default, [200/2], 00:02:47, bgp-65000, internal, tag 65000, segid: 303030 tunnelid: 0xa
```

テナントVRFからデフォルトVRF

この例では、デフォルトVRFにリークされる予定のテナントa VRFのVXLAN経由でルート192.168.10.11をBorder VTEP(BL)が受信しています。

ルーティングテーブルの確認

```
BL# sh ip route 192.168.10.11 vrf tenant-a
IP Route Table for VRF "tenant-a"
'*' denotes best ucast next-hop
'**' denotes best mcast next-hop
'[x/y]' denotes [preference/metric]
'%<string>' in via output denotes VRF <string>

192.168.10.11/32, ubest/mbest: 1/0
*via 172.16.100.10%default, [200/0], 01:15:04, bgp-65000, internal, tag 65000, segid: 303030 tunnelid:
```

フィルタルート

NXOSでは、ルートマップはルートをフィルタリングおよび再配布するためのパラメータとして必要です。この例では、プレフィックス172.16.120.55/32がフィルタリングされます。

設定

	コマンドまたはアクション	目的
手順 1	BL#端末の設定 Enter configuration commands, one per line.CNTL/Z で終了します。	コンフィギュレーションモードを開始します。
手順 2	BL(config)# ip prefix-list VXLAN-VRF-Tenant-to-default permit 192.168.10.11/32	プレフィックスリスト照合ホストを作成します。

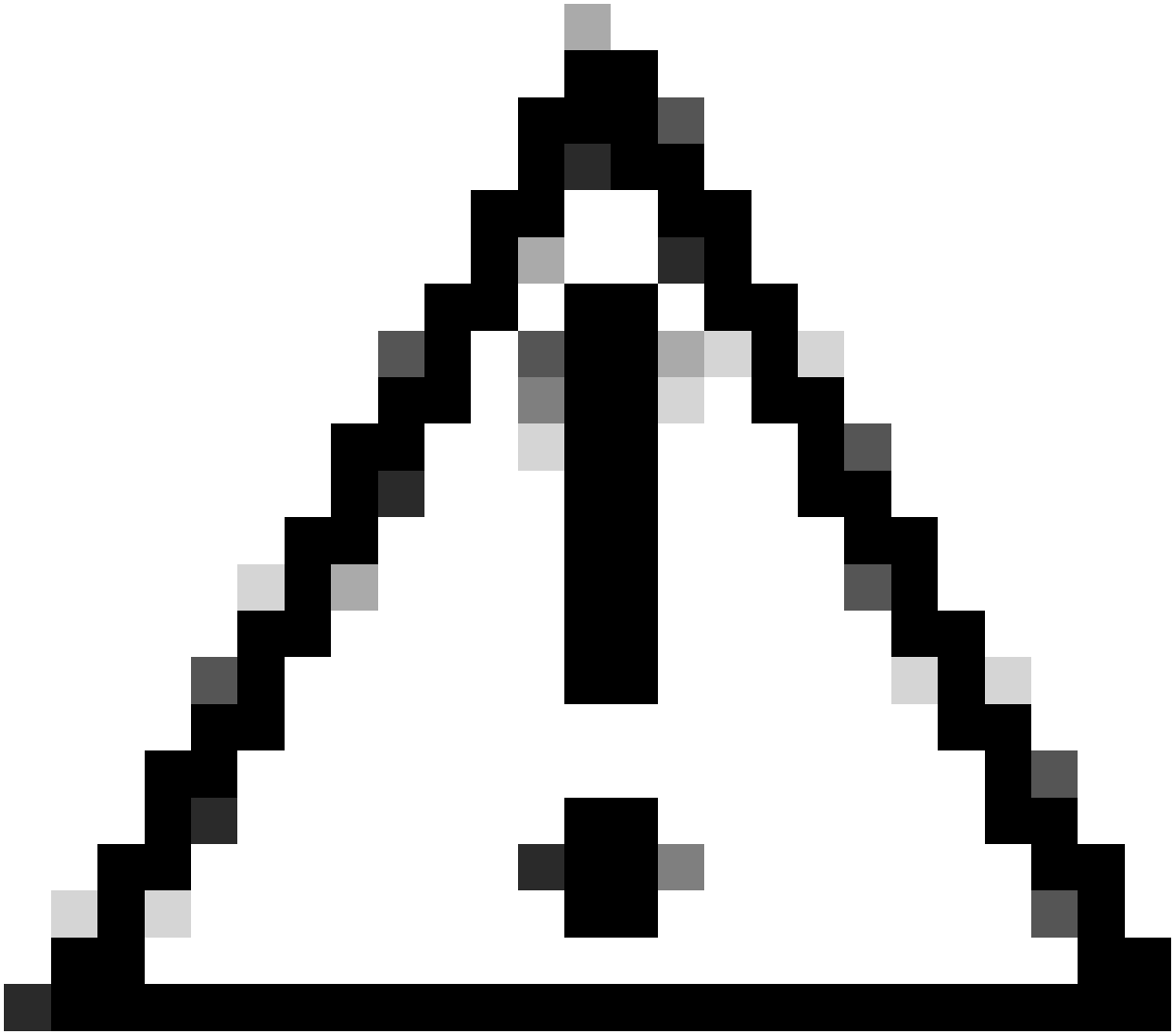
手順 3	BL(config)# route-map VXLAN-VRF-Tenant-to-default	ルートマップを作成します。
手順 4	BL(config-route-map)# match ip address prefix-list VXLAN-VRF-Tenant-to-default	ステップ2で作成したmatch prefix-list。

テナントaのVRFからデフォルトVRFへのルートのエクスポート

ルートはすでにBGP L2VPNプロセスにあるため、VRFのデフォルトにのみエクスポートする必要があります。

設定

	コマンドまたはアクション	目的
手順 1	BL#端末の設定 Enter configuration commands, one per line.CNTL/Z で終了します。	コンフィギュレーションモードを開始します。
手順 2	BL(config)# vrf context tenant-a	VRF設定を開始します。
手順 3	BL(config-vrf)# address-family ipv4 unicast	VRF address-family IPV4と入力します。
手順 4	BL(config-vrf-af-ipv4)# export vrf default map VXLAN-VRF-Tenant-to-default allow-vpn	テナントVRFからデフォルトVRFへのルートのエクスポートによるVPNの許可



注意：デフォルトでは、デフォルト以外のVRFからデフォルトのVRFにエクスポートできるIPプレフィックスの最大数は1000ルートです。この値は、VRF address-family IPV4: export vrf default <number of prefixes> map <route-map name> allow-vpnの下のコマンドで変更できます。

手順の概要

1. configure terminal
2. ip prefix-list VXLAN-VRF-Tenant-to-default permit 192.168.10.11/32を設定する
3. ルートマップVXLAN-VRF – テナントとデフォルト
4. match ip address prefix-list VXLAN-VRF-Tenant-to-default
5. vrfコンテキストテナントa
6. address-family ipv4 unicast
7. vrf default map VXLAN-VRF-Tenant-to-default allow-vpnのエクスポート

確認

デフォルトVRFでルートがBGP IPV4アドレスファミリにインポートされることを確認する

```
BL(config-router-vrf-neighbor)# sh ip bgp 192.168.10.11
BGP routing table information for VRF default, address family IPv4 Unicast
BGP routing table entry for 192.168.10.11/32, version 55
Paths: (1 available, best #1)
Flags: (0x8000001a) (high32 00000000) on xmit-list, is in urib, is best urib route, is in HW

Advertised path-id 1
Path type: internal, path is valid, is best path, no labeled nexthop, in rib
Imported from 172.16.0.5:3:192.168.10.11/32 (VRF tenant-a)
Original source: 172.16.100.1:32777:[2]:[0]:[0]:[48]:[0027.e380.6059]:[32]:[192.168.10.11]/272
AS-Path: NONE, path sourced internal to AS
172.16.100.10 (metric 45) from 10.104.11.1 (192.168.0.11)
Origin IGP, MED not set, localpref 100, weight 0
Received label 101010 303030
Extcommunity: RT:65000:101010 RT:65000:303030 SOO:172.16.100.10:0 ENCAP:8
Router MAC:70db.9855.f52f
Originator: 172.16.100.1 Cluster list: 192.168.0.11

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

ルートがデフォルトのVRFルーティングテーブルにインポートされることを確認します。

```
BL(config-router-vrf-neighbor)# show ip route 192.168.10.11
IP Route Table for VRF "default"
'*' denotes best ucast next-hop
'**' denotes best mcast next-hop
'[x/y]' denotes [preference/metric]
'%<string>' in via output denotes VRF <string>

192.168.10.11/32, ubest/mbest: 1/0
*via 172.16.100.10, [200/0], 00:03:51, bgp-65000, internal, tag 65000, segid: 303030 tunnelid: 0xac1064

Tenant-VRF to Default VRF
```

テナントVRFからテナントVRF

この例では、nexus LEAFは、VRFテナントBにリークされるルート172.16.120.55/32テナントaを受信します

ルーティングテーブルの確認

```
show ip route 172.16.120.55/32 vrf tenant-a
IP Route Table for VRF "tenant-a"
'*' denotes best ucast next-hop
'**' denotes best mcast next-hop
```

```
'[x/y]' denotes [preference/metric]
'%<string>' in via output denotes VRF <string>

172.16.120.55/32, ubest/mbest: 1/0
*via 172.16.0.5%default, [200/2], 4d02h, bgp-65000, internal, tag 65000, segid: 303030 tunnelid: 0xac10
```

フィルタルート

ルートをフィルタリングするには、2つのステップが必要です。VRF間のフィルタリングは、ルートターゲット(RT)経由で行われ、RTは<BGP Process ID>:L3VNI ID>によって準拠され、特定のサブネットをフィルタリングします。2番目の手順を使用しないと、送信元VRFからのすべてのルートが宛先VRFにリークされます。

ルートターゲットの識別

<#root>

```
LEAF# show nve vni
<Snipped>
Interface VNI Multicast-group State Mode Type [BD/VRF] Flags
--------- -------- ----------------- ----- ---- ------------------ -----
nve1 50500 n/a Up CP L3 [tenant-b]
nve1 101010 224.10.10.10 Up CP L2 [10]
nve1 202020 224.10.10.10 Up CP L2 [20]
nve1
```

303030

```
 n/a Up CP L3 [
```

tenant-a

```
]

LEAF# show run bgp | include ignore-case router
router bgp
```

65000

```
router-id 172.16.0.2
```

この例では、ルートターゲットは65000:303030 であり、ルート172.16.120.55/32はフィルタリングされます。

設定

	コマンドまたはアクション	目的

手順 1	LEAF# configure terminal Enter configuration commands, one per line.CNTL/Z で終了します。	コンフィギュレーションモードを開始します。
手順 2	LEAF(config)# ip prefix-list filter-tenant-a-to-tenant-b permit 172.16.120.55/32	プレフィックスリスト照合ホストを作成します。
手順 3	LEAF(config)# route-map tenantA-to-tenantB	ルートマップを作成します。
手順 4	LEAF(config-route-map)# match ip address prefix-listfilter-tenant-a-to-tenant-b	ステップ2で作成した match prefix-list。

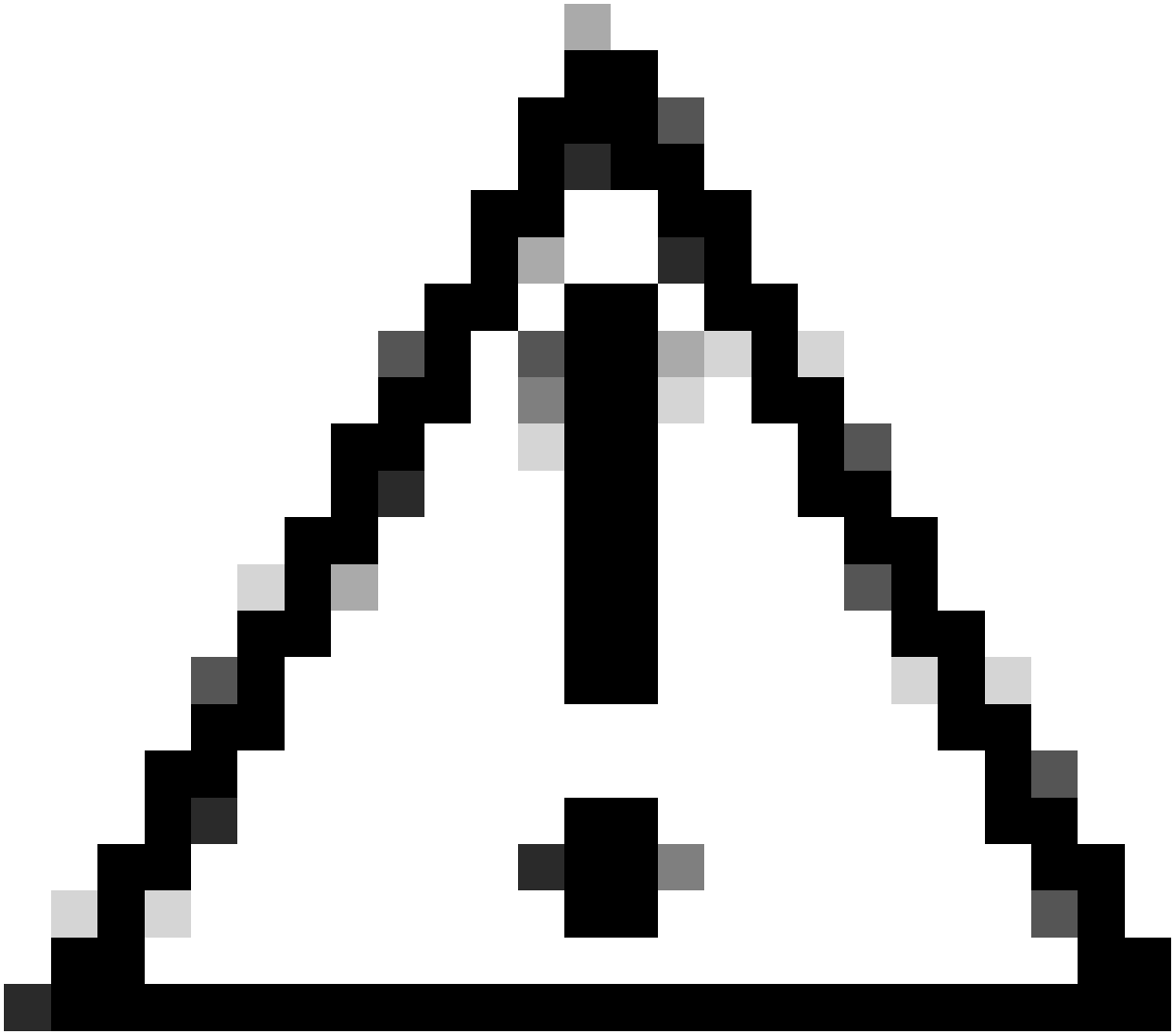
テナントa VRFからテナントa VRFへのルートのインポート

RTが特定され、フィルタリングが設定されると、ルートを宛先VRF（テナントB）にインポートできます

設定

	コマンドまたはアクション	目的
手順 1	LEAF# configure terminal Enter configuration commands, one per line.CNTL/Z で終了します。	コンフィギュレーションモードを開始します。
手順 2	LEAF(config)# vrf context tenant-b	VRF設定を開始します。
手順 3	LEAF(config-vrf)# address-family ipv4 unicast	VRF address-family IPV4と

		入力します。
手順 4	LEAF(config-vrf-af-ipv4)# import map tenantA-to-tenantB	ルートマップでフィルタリングされたルートのインポート
手順 5	LEAF(config-vrf-af-ipv4)# route-target import 65000:303030	ルートターゲットのインポート
手順 6	LEAF(config-vrf-af-ipv4)# route-target import 65000:303030 evpn	ルートターゲットevpnのインポート



注意：インポートマップを使用しないと、すべてのルートが元のVRFからターゲットのVRFに漏出される可能性があります。import mapを使用すると、漏出するルートを制御できます。

手順の概要

1. configure terminal
2. ip prefix-list filter-tenant-a-to-tenant-b permit 172.16.120.55/32を設定します。
3. テナントAからテナントBへのルートマップ
4. ipアドレスprefix-listfilter-tenant-aとtenant-bの照合
5. vrfコンテキストテナントb
6. address-family ipv4 unicast
7. テナントAからテナントBへのマップのインポート
8. ルートターゲットインポート65000:303030
9. ルートターゲットインポート65000:303030 evpn

確認

テナントbのVRFでルートがBGPにインポートされていることを確認します。

```
LEAF(config-vrf-af-ipv4)# show ip bgp 172.16.120.55/32 vrf tenant-b
BGP routing table information for VRF tenant-b, address family IPv4 Unicast
BGP routing table entry for 172.16.120.55/32, version 311
Paths: (1 available, best #1)
Flags: (0x8008021a) (high32 00000000) on xmit-list, is in urib, is best urib route, is in HW
vpn: version 456, (0x00000000100002) on xmit-list

Advertised path-id 1, VPN AF advertised path-id 1
Path type: internal, path is valid, is best path, no labeled nexthop, in rib
Imported from 172.16.0.5:3:[5]:[0]:[0]:[32]:[172.16.120.55]/224
AS-Path: NONE, path sourced internal to AS
172.16.0.5 (metric 45) from 10.101.11.1 (192.168.0.11)
Origin incomplete, MED 2, localpref 100, weight 0
Received label 303030
Extcommunity: RT:65000:303030 ENCAP:8 Router MAC:20cf.ae54.fa3b
OSPF RT:0.0.0.0:0:0
Originator: 172.16.0.5 Cluster list: 192.168.0.11

VRF advertise information:
Path-id 1 not advertised to any peer

VPN AF advertise information:
Path-id 1 not advertised to any peer
```

テナントBのVRFのルーティングテーブルにルートがインポートされていることを確認します。

```
LEAF# show ip route 172.16.120.55/32 vrf tenant-b
IP Route Table for VRF "tenant-b"
'*' denotes best ucast next-hop
'**' denotes best mcast next-hop
'[x/y]' denotes [preference/metric]
'%<string>' in via output denotes VRF <string>

172.16.120.55/32, ubest/mbest: 1/0
*via 172.16.0.5%default, [200/2], 00:00:08, bgp-65000, internal, tag 65000, segid: 303030 (Asymmetric)
```

翻訳について
シスコは世界中のユーザにそれぞれの言語でサポート コンテンツを提供するために、機械と人による翻訳を組み合わせて、本ドキュメントを翻訳しています。ただし、最高度の機械翻訳であっても、専門家による翻訳のような正確性は確保されません。シスコは、これら翻訳の正確性について法的責任を負いません。原典である英語版（リンクからアクセス可能）もあわせて参照することを推奨します。