

# Configurazione di BGP VRF Auto RD RT per EVPN sugli switch Catalyst serie 9000

## Sommario

---

[Introduzione](#)

[Prerequisiti](#)

[Requisiti](#)

[Componenti usati](#)

[Premesse](#)

[Terminologia](#)

[Configurazione](#)

[VRF RD-auto globale](#)

[Per configurazione VRF rd-auto](#)

[RD statico misto e RD automatico](#)

[Famiglia di indirizzi BGP IPv4 Vrf e Ipv6 Vrf](#)

[Verifica](#)

[Foglia](#)

[Risoluzione dei problemi](#)

[Debug](#)

[Interoperabilità Catalyst e Nexus](#)

[Problema](#)

[Correzione](#)

[Informazioni correlate](#)

---

## Introduzione

Questo documento descrive l'interfaccia CLI di semplificazione EVPN per Auto RD e Auto RT BGP VRF in EVPN sugli switch Catalyst serie 9000.

## Prerequisiti

### Requisiti

Cisco raccomanda la conoscenza dei seguenti argomenti:

- Configurazione BGP di base
- Configurazione VRF di base
- Configurazione EVPN di base

### Componenti usati

Le informazioni fornite in questo documento si basano sulle seguenti versioni software e hardware:

- Catalyst 9300
- Catalyst 9400
- Catalyst 9500
- Catalyst 9600
- Cisco IOS® XE 17.12.1 e versioni successive

Le informazioni discusse in questo documento fanno riferimento a dispositivi usati in uno specifico ambiente di emulazione. Su tutti i dispositivi menzionati nel documento la configurazione è stata ripristinata ai valori predefiniti. Se la rete è operativa, valutare attentamente eventuali conseguenze derivanti dall'uso dei comandi.

## Premesse

Le distribuzioni EVPN di layer 3 prevedono configurazioni VRF con numerose opzioni di configurazione, tra cui, a titolo esemplificativo, identificatore di percorso (RD) e destinazione di percorso (RT).

- Prima dell'introduzione della funzionalità Auto RD RT della VRF BGP sarebbero necessarie almeno 5 linee di configurazione (1 per RD, 4 per RT) per configurare una particolare VRF per l'utilizzo della VPN BGP.
- Con BGP VRF Auto RD Auto RT è possibile ottenere questo risultato solo con 2 linee (potenzialmente una linea per VRF se è abilitato VRF rd-auto globale).
- Non esiste alcuna differenza funzionale tra Desktop remoto automatico e Desktop remoto statico. Ogni Desktop remoto deve essere univoco all'interno di un router o di uno switch specifico.
- La differenza funzionale tra Auto RT e statico RT è che Auto RT è solo uno e lo stesso per l'importazione e l'esportazione, regolare e cucitura, mentre RT statico può essere configurato da zero a molti.
- Inoltre, Auto RT può coesistere con RT statico all'interno di qualsiasi VRF (è possibile configurare Auto RT in aggiunta a RT statico esistente prima di questa funzione).

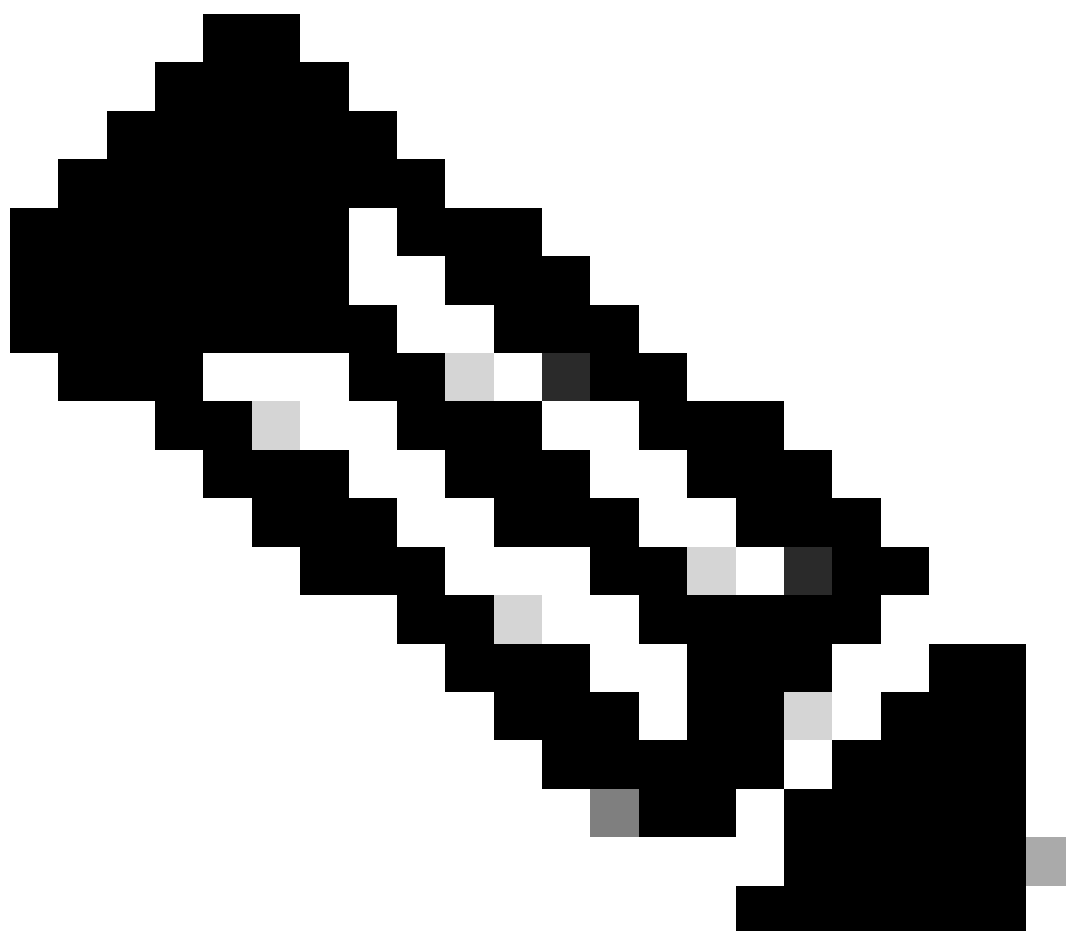
Il RD automatico è costituito dall'ID del router BGP più un numero univoco generato internamente. Ad esempio, se l'ID del router BGP è 192.168.1.1, il RD automatico sarà simile a "192.168.1.1:1".

- L'auto RT è costituita dal numero BGP AS più il vnid da configurare. Ad esempio, se il numero BGP AS è 65000 e il ID virtuale è configurato come 123, il valore di RT automatico sarà "6500:123".
- Questo vale sia per le destinazioni di route regolari di importazione ed esportazione che per quelle di stitching.
- Se BGP AS è di 4 byte, viene utilizzato AS\_TRANS, che è 23456.

La possibilità di semplificare la configurazione è fortemente auspicabile (se non necessaria) affinché l'installazione sia fattibile ed è già stata ampiamente adottata per il fabric BGP EVPN. Questa funzione è desiderabile per EVPN, in quanto consente di evitare la scrittura e la

manutenzione di configurazioni estese e complesse nelle topologie Spine-Leaf in cui molti VRF sono configurati in una particolare foglia.

---



Nota: questa funzione introduce nuove CLI.

---

## Terminologia

VRF	Inoltro routing virtuale	Definisce un dominio di routing di livello 3 che deve essere separato da altri VRF e da un dominio di routing IPv4/IPv6 globale
AF	Famiglia indirizzi	Definisce quali prefissi di tipo e informazioni di routing sono handle BGP
AS	Sistema autonomo	Set di prefissi IP instradabili Internet appartenenti a una rete o a un insieme di reti gestite, controllate e supervisionate da una singola

		entità o organizzazione
RD	Distintore route	Consenti a BGP di distinguere un prefisso da un altro in VRF diverse
RT	Destinazione route	Le destinazioni dei cicli di lavorazione vengono utilizzate per vincolare gli aggiornamenti dei cicli. Determina quali prefissi possono essere importati dal dispositivo
EVPN	Ethernet Virtual Private Network	L'estensione che consente a BGP di trasportare le informazioni MAC di layer 2 e IP di layer 3 è EVPN e utilizza il protocollo MP-BGP (Multi-Protocol Border Gateway Protocol) per distribuire le informazioni sulla raggiungibilità relative alla rete di sovrapposizione VXLAN.
VXLAN	LAN virtuale estendibile (LAN)	La VXLAN è progettata per superare i limiti intrinseci delle VLAN e dell'STP. Si tratta di uno standard IETF [RFC 7348] proposto per fornire gli stessi servizi di rete Ethernet di layer 2 delle VLAN, ma con una maggiore flessibilità. A livello funzionale, è un protocollo di incapsulamento MAC-in-UDP che viene eseguito come sovrapposizione virtuale su una rete sottostante di layer 3.

## Configurazione

### VRF RD-auto globale

```
<#root>
```

```
Leaf-01#
```

```
sh run | include vrf rd-auto
```

```
vrf rd-auto
```

```
<-- Enable Auto RD for all the VRFs
```

```
Leaf-01#
```

```
sh run | section vrf definition blue
```

```
vrf definition blue
```

```
vnid 123 evpn-instance
```

```
<-- Enable Auto RT
```

```
!
```

```
address-family ipv4
```

```
<-- address-family needs to be specified
```

```
route-target 100:123
```

```
<-- Optionally can have static route-target as req
```

```
exit-address-family
```

```
!
```

## Per configurazione VRF rd-auto

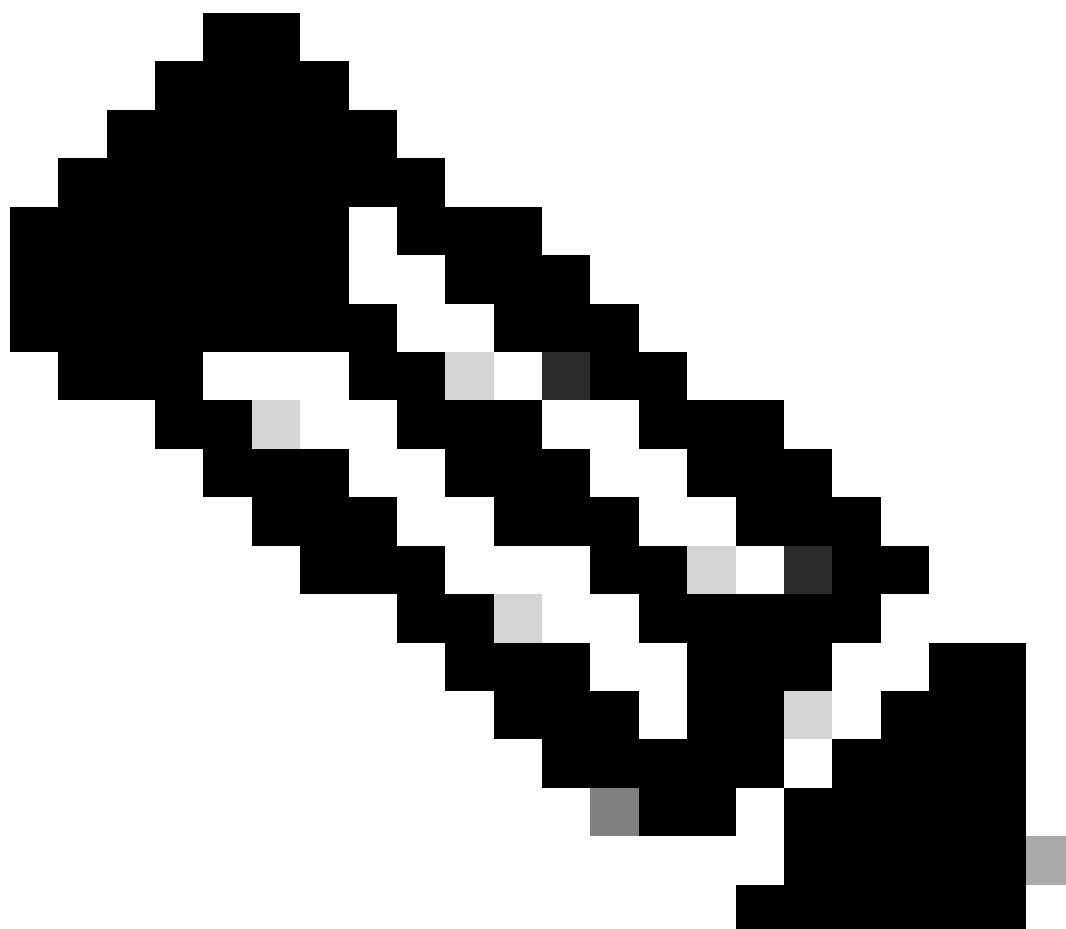
```
<#root>
```

```
Leaf-01#
```

```
sh run | section vrf definition green
```

```
vrf definition green
rd-auto                               <-- Enable Auto RD for this VRF green
vniid 35 evpn-instance                <-- Enable Auto RT
!
address-family ipv4                   <-- address-family needs to be specified
exit-address-family
!
address-family ipv6
exit-address-family
```

---

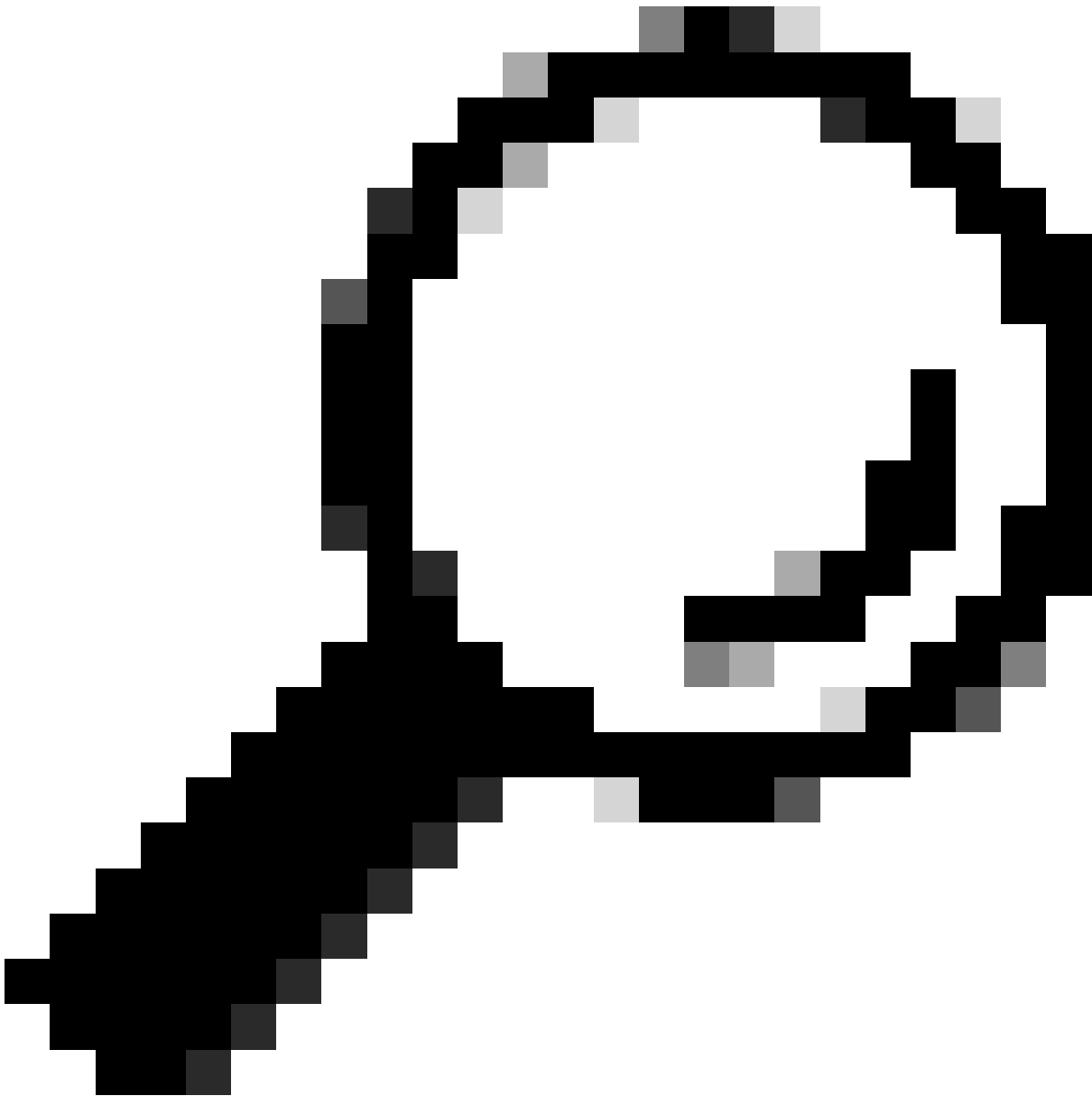


---

Nota: è possibile disporre di Desktop remoto statico e automatico per VRF diversi, ma gli RD statici NON devono avere lo stesso RD effettivo dell'RD automatico se quest'ultimo viene assegnato per primo.

---

---



Suggerimento: se si elimina il RD statico, verrà eliminata la configurazione delle route-target configurate nei VRF e nelle famiglie di indirizzi VRF BGP IPv4 e/o IPv6 (e la configurazione associata sottostante). L'eliminazione di un Desktop remoto automatico avrebbe pertanto un comportamento simile. Si consiglia di non attivare l'eliminazione del RD a meno che non sia assolutamente necessario. Una modifica dello stato di Desktop remoto, ovvero un'eliminazione dello stato esistente, statico o automatico, e quindi l'aggiunta di un nuovo Desktop remoto, statico o automatico, è costosa e richiede un tempo di ritardo per l'esecuzione del comando

---

## RD statico misto e RD automatico

```
<#root>

vrf rd-auto
vrf definition green                                <-- This VRF green uses auto RD
  vnid 35 evpn-instance
!
  address-family ipv6
  exit-address-family
vrf definition red                                  <-- This VRF red uses static RD
  rd-auto disable
  rd 100:1
!
  address-family ipv4
    route-target export 100:1
    route-target import 100:1
    route-target export 100:1 stitching
    route-target import 100:1 stitching
  exit-address-family
```

## Famiglia di indirizzi BGP IPv4 Vrf e Ipv6 Vrf

(L'esempio di configurazione è un riassunto della feature esistente)

```
<#root>

Leaf-01#

show run | sec r bgp

router bgp 65000

<-- Required for Auto RT

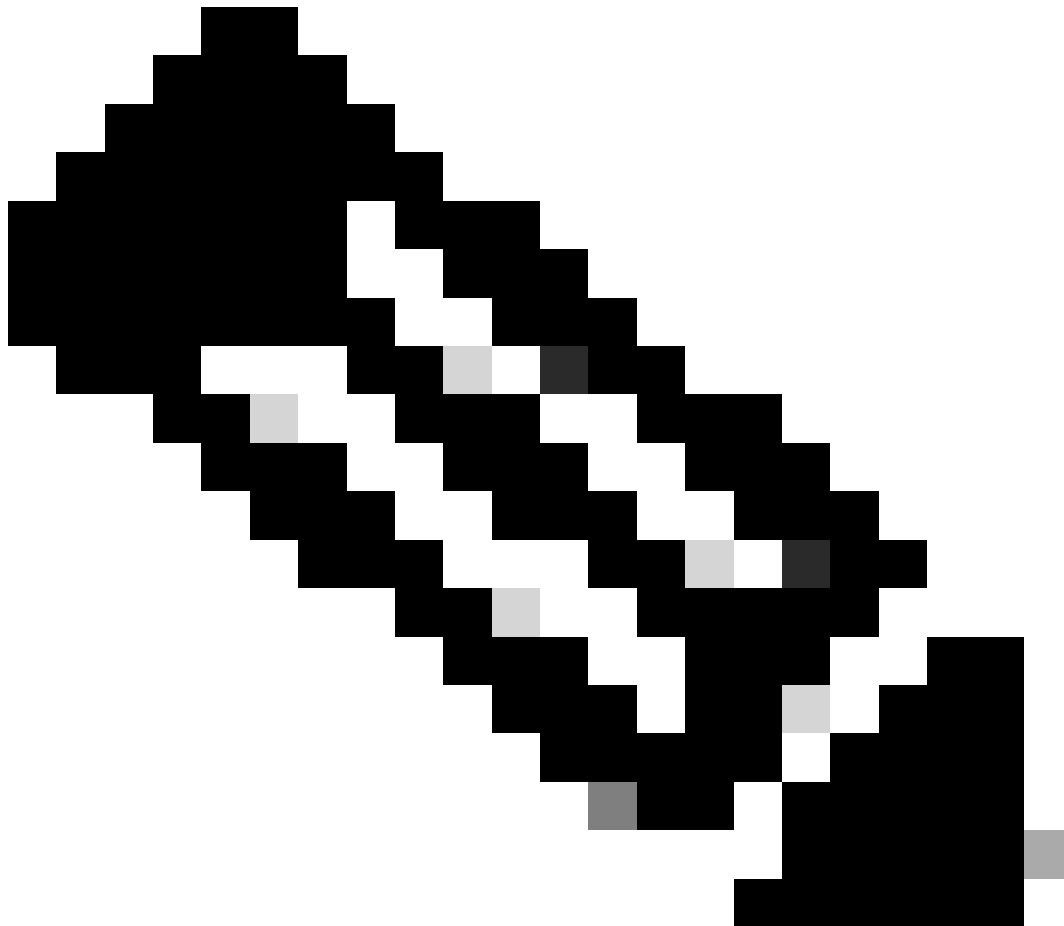
bgp router-id 192.168.1.1

<-- Required for Auto RD

bgp log-neighbor-changes
no bgp default ipv4-unicast
neighbor 192.168.1.2 remote-as 65000
neighbor 192.168.1.2 update-source Loopback0
neighbor 192.168.1.3 remote-as 65001
neighbor 192.168.1.3 update-source Loopback0
!
address-family ipv4 vrf green
  advertise l2vpn evpn
  redistributed connected
exit-address-family
!
```

```
address-family ipv6 vrf green
advertise l2vpn evpn
redistribute connected
exit-address-family
```

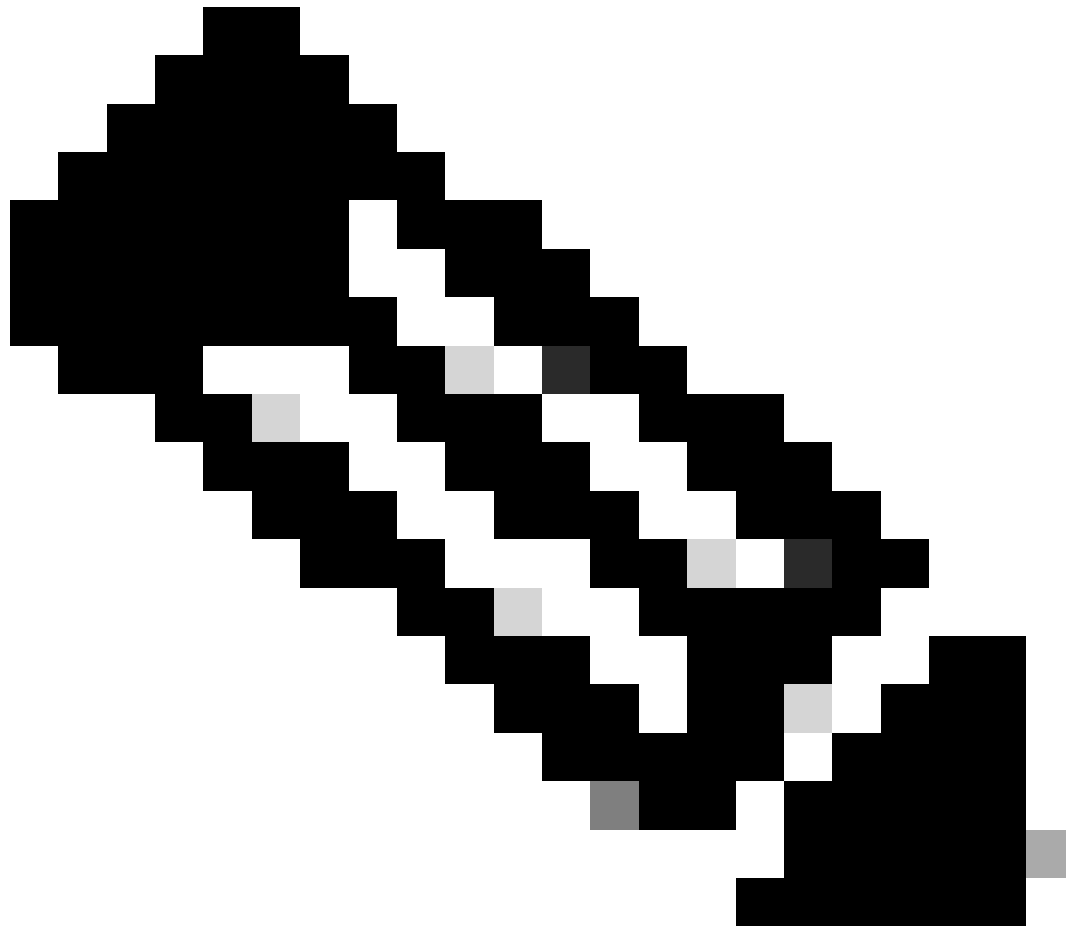
---



Nota: la configurazione dell'altro riflettore di instradamento della curva guida è la stessa, quindi non vengono ripetute in questa sezione

---





Nota: altri fogli EVPN possono utilizzare la configurazione statica di RD o RT. Finché l'RT corrisponde, i prefissi EVPN possono essere importati o esportati l'uno verso l'altro.

---

## Verifica

### Foglia

Verificare la foglia per disporre di Desktop remoto automatico

```
<#root>
```

```
VTEP1#
```

```
show vrf blue
```

Name	Default RD	Protocols	Interfaces
blue	192.168.1.1:1(auto)	ipv4	V134 Lo101

Et1/1  
V14  
V115

<#root>

VTEP1#

show vrf green

Name	Default RD	Protocols	Interfaces
green	192.168.1.1:2(auto)	ipv6	Lo102 Et1/2 V15 V113

<#root>

VTEP1#

show vrf detail blue

VRF blue (VRF Id = 2); default RD 192.168.1.1:1(auto); default VPNID

New CLI format, supports multiple address-families

vnid: 123 evpn-instance vni 35000 core-vlan 34

Flags: 0x180C

Interfaces:

V134	Lo101	Et1/1
V14	V115	

Address family ipv4 unicast (Table ID = 0x2):

Flags: 0x0

Export VPN route-target communities

RT:100:123 RT:65000:123 (auto)

Import VPN route-target communities

RT:100:123 RT:65000:123 (auto)

Export VPN route-target stitching communities

RT:65000:123 (auto)

Import VPN route-target stitching communities

RT:65000:123 (auto)

No import route-map

No global export route-map

No export route-map

VRF label distribution protocol: not configured

VRF label allocation mode: per-prefix

Address family ipv6 unicast not active

Address family ipv4 multicast not active

Address family ipv6 multicast not active

<#root>

VTEP1#

show vrf detail green

VRF green (VRF Id = 4); default RD 192.168.1.1:2(auto); default VPNID

New CLI format, supports multiple address-families

```
vnid: 35 evpn-instance
Flags: 0x380C
Interfaces:
  Lo102          Et1/2          V15
  V113
Address family ipv4 unicast not active
Address family ipv6 unicast (Table ID = 0x1E000002):
  Flags: 0x0
  Export VPN route-target communities
    RT:65000:35 (auto)
  Import VPN route-target communities
    RT:65000:35 (auto)
  Export VPN route-target stitching communities
    RT:65000:35 (auto)
  Import VPN route-target stitching communities
    RT:65000:35 (auto)
  No import route-map
  No global export route-map
  No export route-map
  VRF label distribution protocol: not configured
  VRF label allocation mode: per-prefix
Address family ipv4 multicast not active
Address family ipv6 multicast not active
```

## Risoluzione dei problemi

### Debug

In caso di problemi con VRF auto RD auto RT, è possibile utilizzare i debug per visualizzare ulteriori informazioni sul problema

Abilita debug rilevanti

```
<#root>
```

```
Leaf-01#
```

```
debug ip bgp autordrt
```

```
Leaf-01#
```

```
debug vrf create
```

```
Leaf-01#
```

```
debug vrf delete
```

Visualizza informazioni di debug

```
<#root>
```

```
VTEP1#
```

```
show debug
```

```
VRF Manager:
```

```
VRF creation debugging is on
```

```
VRF deletion debugging is on
```

```
Packet Infra debugs:
```

```
Ip Address Port
```

```
-----|-----  
IP routing:
```

```
BGP auto rd rt debugging is on
```

Osservare i debug prodotti in ciascuna fase della configurazione

```
<#root>
```

```
Leaf-01(config)#
```

```
vrf definition test
```

```
*Jun 26 08:19:44.173: LID: Get id @0x7F4414FE4A18 - current A [1..2705] (checking enabled)
```

```
*Jun 26 08:19:44.173: LID: AVAIL (verified) - id A
```

```
*Jun 26 08:19:44.173: vrfmgr: VRF test: Created vrf_rec with vrfid 0xA
```

```
*Jun 26 08:19:44.173: BGP: VRF config event of
```

```
rd-auto change for vrf test
```

```
*Jun 26 08:19:44.173: BGP-VPN: bgp vpn global
```

```
rd-auto for vrf test assigns rd of 192.168.1.1:6
```

```
*Jun 26 08:19:44.173: BGP: VRF config event of
```

```
vnid change for vrf test
```

```
Leaf-01(config-vrf)#
```

```
vnid 246 evpn-instance
```

```
% vnid 246 evpn-instance auto (vni 0 core-vlan 0) is configured in "vrf test"
```

```
*Jun 26 08:20:03.466: BGP: VRF config event of
```

```
vnid change for vrf test
```

```
Leaf-01(config-vrf)#
```

```
address-family ipv4
```

```
*Jun 26 08:20:12.276: vrfmgr: VRF test ipv4 unicast: Received topology create notification
```

```
*Jun 26 08:20:12.276: vrfmgr: VRF test ipv4 multicast: Received topology create notification
```

```
*Jun 26 08:20:12.276: vrfmgr: VRF test ipv4 unicast:
```

```
Created vrf_sub_rec with vrfid 0xA, tableid 0xA
```

```
*Jun 26 08:20:12.276: BGP: VRF config event of vnid change for vrf test
```

```
*Jun 26 08:20:12.276: BGP: afi 0 vrf
```

```
test vnid 246 RT assign
```

```
*Jun 26 08:20:12.276: BGP: vrf assign auto import stitching rt for VRF test
*Jun 26 08:20:12.276: BGP: vrf assign auto export stitching rt for VRF test
```

```
Leaf-01(config-vrf-af)#
```

```
address-family ipv6
```

```
*Jun 26 08:20:20.949: vrfmgr: VRF test ipv6 unicast: Received topology create notification
*Jun 26 08:20:20.949: vrfmgr: VRF test ipv6 multicast: Received topology create notification
*Jun 26 08:20:20.949: vrfmgr: VRF test ipv6 unicast:
```

```
Created vrf_sub_rec with vrfid 0xA, tableid 0x1E000004
```

```
*Jun 26 08:20:20.949: BGP: VRF config event of vnid change for vrf test
*Jun 26 08:20:20.949: BGP:
```

```
afi 0 vrf test vnid 246 RT assign
```

```
*Jun 26 08:20:20.949: BGP: vrf assign auto import stitching rt for VRF test
*Jun 26 08:20:20.949: BGP: vrf assign auto export stitching rt for VRF test
*Jun 26 08:20:20.949: BGP:
```

```
afi 1 vrf test vnid 246 RT assign
```

```
*Jun 26 08:20:20.949: BGP: vrf assign auto import stitching rt for VRF test
*Jun 26 08:20:20.949: BGP: vrf assign auto export stitching rt for VRF test
```

```
Leaf-01(config-vrf-af)#
```

```
do sh vrf detail test
```

```
VRF test (VRF Id = 10)
```

```
; default
```

```
RD 192.168.1.1:6(auto)
```

```
; default VPNID
```

```
<-- VRF ID = 10 (hex 0xA) | auto RD assigned matches debug "assigns rd of 192.168.1.1:6"
```

```
New CLI format, supports multiple address-families
```

```
vnid: 246
```

```
evpn-instance
  Flags: 0x180C
  No interfaces
Address family ipv4 unicast (Table ID = 0xA):
  Flags: 0x0
  Export VPN route-target communities
    RT:65000:246 (auto)
  Import VPN route-target communities
    RT:65000:246 (auto)
  Export VPN route-target stitching communities
    RT:65000:246 (auto)
  Import VPN route-target stitching communities
    RT:65000:246 (auto)
  No import route-map
  No global export route-map
  No export route-map
  VRF label distribution protocol: not configured
  VRF label allocation mode: per-prefix
Address family ipv6 unicast
(Table ID = 0x1E000004)
```

```
:
<-- ID matches debug
"
Created vrf_sub_rec with vrfid 0xA, tableid 0x1E000004"

Flags: 0x0
Export VPN route-target communities
  RT:65000:246 (auto)
Import VPN route-target communities
  RT:65000:246 (auto)
Export VPN route-target stitching communities
  RT:65000:246 (auto)
Import VPN route-target stitching communities
  RT:65000:246 (auto)
No import route-map
No global export route-map
No export route-map
VRF label distribution protocol: not configured
VRF label allocation mode: per-prefix
Address family ipv4 multicast not active
Address family ipv6 multicast not active

Leaf-01(config-vrf-af)#
do sh run vrf test

Building configuration...

Current configuration : 145 bytes
vrf definition test
 vnid 246 evpn-instance
 !
 address-family ipv4
 exit-address-family
 !
 address-family ipv6
 exit-address-family
```

## Interoperabilità Catalyst e Nexus

### Problema

Per impostazione predefinita, Nexus assegna route-target basati su vni (ASN:VNI), mentre Catalyst assegna route-target basati su evi (ASN:EVI).

Se route-target non corrisponde, è possibile osservare i seguenti sintomi:

- Viene stabilita la connessione BGP per l'EVPN L2VPN e le route di tipo 3 sono visibili nella tabella BGP
- Peering NVE non stabilito
- L'adiacenza del tunnel rimane incompleta

### Correzione

Per risolvere il problema di interoperabilità, sono disponibili due opzioni

1. Configurare le destinazioni di instradamento manuali su un lato in modo che corrispondano
2. Configurare C9500 per assegnare route-target basati su vni utilizzando 'route-target auto vni'

Applicare queste cli (per l'opzione 2) nella sezione l2vpn evpn

```
<#root>
```

```
address-family l2vpn evpn
```

```
rewrite-evpn-rt-asn <--->
```

## Informazioni correlate

- [Guida alla configurazione della VXLAN BGP VPN, Cisco IOS XE Dublin 17.11.x \(switch Catalyst 9500\)](#)
- [Documentazione e supporto tecnico – Cisco Systems](#)

## Informazioni su questa traduzione

Cisco ha tradotto questo documento utilizzando una combinazione di tecnologie automatiche e umane per offrire ai nostri utenti in tutto il mondo contenuti di supporto nella propria lingua. Si noti che anche la migliore traduzione automatica non sarà mai accurata come quella fornita da un traduttore professionista. Cisco Systems, Inc. non si assume alcuna responsabilità per l'accuratezza di queste traduzioni e consiglia di consultare sempre il documento originale in inglese (disponibile al link fornito).