Use VNID to Auto Provision L3VNI, Core-SVI, Core-vlan on Catalyst 9000 Switches

Contents

Introduction

This document describes auto provision options for the L3VNI, core-SVI and core-vlan using VNID on Catalyst 9000 Series Switches.

Prerequisites

Requirements

Cisco recommends that you have knowledge of these topics:

- Basic BGP configuration
- Basic VRF configuration
- Basic EVPN configuration

> **Note**: Please refer to the [Configure BGP VRF Auto RD Auto RT for EVPN](#) for more detailed information on EVPN CLI simplification.

Components Used

The information in this document is based on these software and hardware versions:

- Catalyst 9300
- Catalyst 9400
- Catalyst 9500
- Catalyst 9600
- Cisco IOS® XE 17.13.1 and later

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If your network is live, ensure

that you understand the potential impact of any command.

Background Information

Layer 3 EVPN deployments involve many configuration options, including but not limited to, Layer 3 VNI association of VRF to NVE interface, core SVI, and core-vlan.

- Before the introduction of the Auto provision of L3VNI and core-vlan feature at least 7 configuration lines (1 for L3VNI association with nve interface, 2 for core SVI, 4 for core-vlan) would be required to set up a particular VRF for BGP EVPN VxLAN usage.
- **With Auto provision of L3VNI and core-vlan feature this can be achieved with only 1 line, particularly at the optional parameters of vnid command (which is required for auto RT**).
- There is no functional difference between Auto provision of L3VNI, core-SVI, core-vlan and static L3VNI, core-SVI, core-vlan. Each VRF must be uniquely associated with one L3VNI, core-SVI, and core-vlan.
- The legal range of L3VNI is from 4096 to 16777215 (inclusively). Using the option of "vni auto", the vnid value need to be within this range.
- The legal range of core-vlan is from 1 to 4094.
- Also the Auto provision of L3VNI and core-SVI/core-vlan can co-exist with static RT within any particular VRF (you can configure Auto RT in additional to existing static RT prior to this feature).

The ability to simplify the configuration is highly desirable (if not necessary) for the deployment to be feasible, and has already been widely adopted for the BGP EVPN VxLAN fabric. This feature is desirable for EVPN, as it helps avoid the writing and maintenance of extensive and complex configurations in Spine-Leaf topologies where many VRFs are configured in a particular leaf.

Note: This feature introduces new CLIs, and is part of CLI simplification for BGP VRF in EVPN on Catalyst 9000 series switches.

Terminology

VRF	Virtual Routing Forwarding	Defines a layer 3 routing domain that be separated from other VRF and global IPv4/IPv6 routing domain
AF	Address Family	Defines which type prefixes and routing info BGP handles
AS	Autonomous System	A set of Internet routable IP prefixes that belong to a network or a collection of networks that are all managed, controlled and supervised by a single entity or organization
EVPN	Ethernet Virtual Private Network	Extension that allows BGP to transport Layer 2 MAC and Layer 3 IP information is EVPN and uses Multi-Protocol Border Gateway Protocol (MP-BGP) as the protocol to distribute reachability information that pertains to the VXLAN overlay network.
VXLAN	Virtual Extensible LAN (Local Area	VXLAN is designed to overcome the inherent limitations of VLANs and STP. It is a proposed IETF standard [RFC 7348] to provide the same

	Network)	Ethernet Layer 2 network services as VLANs do, but with greater flexibility. Functionally, it is a MAC-in-UDP encapsulation protocol that runs as a virtual overlay on a Layer 3 underlay network.

Configure

L3VNI, Core-SVI, Core-vlan Auto Provision

```
<#root>

Leaf-01#

sh run | section vrf definition blue


vrf definition blue
 vnid 31234 evpn-instance vni auto core-vlan 200       <-- Auto provision the L3VNI of 31234,
                                                       <-- core-SVI and core-vlan 200


Leaf-01

#sh run | section vrf definition green
vrf definition green
 vnid 12 evpn-instance vni 16777215 core-vlan 4094     <-- Provision the L3VNI of 16777215,
                                                       <-- core-SVI and core-vlan 4094
```

> **Note**: The configuration for the other Spine Route Reflector is the same, so are not repeated in this section

---

> **Note**: Other EVPN leaves can use static L3VNI, core-SVI, core-vlan configuration. As long as the L3VNI matches, the data traffic is routed properly.

Verify

Verify the Leaf, to have correct L3VNI association with NVE interface, core-SVI and core-vlan:

```
<#root>

Leaf-01#

show derived-config interface nve1

...
interface nve1
 source-interface Loopback1
 host-reachability protocol bgp
...
 member vni 31234 vrf blue
 member vni 16777215 vrf green
```

```
...
```

```
Leaf-01#
```

show derived-config | section vlan configuration 200

```
vlan configuration 200
 member vni 31234
```

```
Leaf-01#
```

show derived-config | section vlan configuration 4094

```
vlan configuration 4094
 member vni 16777215
```

```
Leaf-01#
```

show derived-config | section interface Vlan200

```
interface Vlan200
 vrf forwarding blue
 ip unnumbered Loopback1
 ipv6 unnumbered Loopback1
 no autostate
```

```
Leaf-01#
```

show derived-config | sec interface Vlan4094

```
interface Vlan4094
 vrf forwarding green
 ip unnumbered Loopback1
 ipv6 unnumbered Loopback1
 no autostate
```

Troubleshoot

Debugs

If there is an issue with VRF auto RD auto RT you can use debugs to see more about the problem

Enable relevant debugs

<#root>

```
Leaf-01#
```

debug ip bgp autordrt

```
Leaf-01#
```

debug vrf create

```
Leaf-01#
```

debug vrf delete

```
Leaf-01#
```

debug l2vpn evpn event detail

Display debug information

<#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

```
EVPN:
```

EVPN detailed events debugging is on

Observe the debugs produced at each configuration step

<#root>

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

vrf definition test

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

vnid 31234 evpn-instance vni auto core-vlan 200

```
% vnid vni (31234) core vlan (200) acquired for "vrf test"
% vnid 31234 evpn-instance vni auto (vni 0 core-vlan 200) is configured in "vrf test"

*Aug 12 02:57:07.414: EVPN: Received interface Vlan200 down event
*Aug 12 02:57:07.414: EVPN: Received interface Vlan200 down event
*Aug 12 02:57:07.415: BGP: VRF config event of vnid change for vrf test
*Aug 12 02:57:07.415: BGP: afi 0 vrf test vnid 31234 RT assign
*Aug 12 02:57:07.415: BGP: vrf assign auto import stitching rt for VRF test
*Aug 12 02:57:07.415: BGP: vrf assign auto export stitching rt for VRF test
*Aug 12 02:57:07.415: BGP: afi 1 vrf test vnid 31234 RT assign
*Aug 12 02:57:07.415: BGP: vrf assign auto import stitching rt for VRF test
*Aug 12 02:57:07.415: BGP: vrf assign auto export stitching rt for VRF test
```

```
*Aug 12 02:57:07.416: EVPN: Received interface Vlan200 down event
*Aug 12 02:57:07.416: EVPN: Received interface Vlan200 down event
*Aug 12 02:57:07.416: EVPN: Received interface Vlan200 down event
*Aug 12 02:57:07.416: EVPN: Received interface Vlan200 up event
*Aug 12 02:57:07.416: EVPN: Received interface Vlan200 up event
*Aug 12 02:57:07.416: EVPN: Received interface Vlan200 down event
*Aug 12 02:57:07.416: EVPN: Received interface Vlan200 down event
*Aug 12 02:57:07.416: EVPN: Received interface Vlan200 down event
*Aug 12 02:57:07.416: EVPN: Received interface Vlan200 up event
*Aug 12 02:57:07.416: EVPN: Received interface Vlan200 up event

*Aug 12 02:57:08.414: %LINEPROTO-5-UPDOWN: Line protocol on Interface Vlan200, changed state to up

*Aug 12 02:57:09.414: %LINK-3-UPDOWN: Interface Vlan200, changed state to up

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

no vnid 31234 vni auto core-vlan 200

```
% vnid 31234 delete for VRF test
% deacquire vnid vni (31234) or core vlan (200) succeed for "vrf test"

*Aug 12 04:46:34.462: EVPN: Received interface Vlan200 down event
*Aug 12 04:46:34.462: EVPN: Received interface Vlan200 down event
*Aug 12 04:46:34.462: EVPN: Received interface Vlan200 down event
*Aug 12 04:46:34.463: EVPN: Received interface Vlan200 down event
*Aug 12 04:46:34.464: BGP: VRF config event of vnid change for vrf test
*Aug 12 04:46:34.464: BGP: afi 0 vrf test vnid config deleted
*Aug 12 04:46:34.464: BGP: vrf de-assign auto import stitching rt for VRF test
*Aug 12 04:46:34.464: BGP: vrf de-assign auto export stitching rt for VRF test
*Aug 12 04:46:34.464: BGP: afi 1 vrf test vnid config deleted

*Aug 12 04:46:34.464: BGP: vrf de-assign auto import stitching rt for VRF test
*Aug 12 04:46:34.464: BGP: vrf de-assign auto export stitching rt for VRF test
```

Related Information

- [BGP EVPN VXLAN Configuration Guide, Cisco IOS XE Dublin 17.11.x (Catalyst 9500 Switches)](#)
- [Technical Support & Documentation - Cisco Systems](#)