Enterprise Advanced Routing and Services

ENARSI 300-410 V1.0 Concentration Exam

1st Edition

Contents at a Glance

Chapter 1 Layer 3 Technologies
Chapter 2 VPN Technologies
Chapter 3 Infrastructure Security
Chapter 4 Infrastructure Services

Table of Contents

About the Author 5

Preface 6

What's available on the CCIEin8Weeks website 7

Chapter 1 Layer 3 Technologies 8

Troubleshoot administrative distance (all routing protocols) 9
Troubleshoot route map for any routing protocol (attributes, tagging, filtering) 10
Troubleshoot loop prevention mechanisms (filtering, tagging, split horizon, route poisoning) 11
Troubleshoot redistribution between any routing protocols or routing sources 13
Troubleshoot manual and auto-summarization with any routing protocol 14
Configure and verify policy-based routing 15
Configure and verify VRF-Lite 16
Describe Bidirectional Forwarding Detection (BFD) 21
Troubleshoot EIGRP (classic and named mode) 22
Troubleshoot OSPF (v2/v3) 25
Network types, area types, and router types 26
Path preference 28
Troubleshoot BGP (Internal and External) 29

Chapter 2 VPN Technologies 36

Describe MPLS operations (LSR, LDP, label switching, LSP) 37

Describe MPLS Layer 3 VPN 38

Configure and verify DMVPN (single hub) 38

Chapter 3 Infrastructure Security 47

Troubleshoot device security using IOS AAA (TACACS+, RADIUS, local database) 48

Troubleshoot router security features 49

IPv4 access control lists (standard, extended, time-based) 49
IPv6 Traffic Filter 50
Unicast reverse path forwarding (uRPF) 51

Troubleshoot control plane policing (CoPP) (Telnet, SSH, HTTP(S), SNMP, EIGRP, OSPF, BGP) 52

Describe IPv6 First Hop security features (RA guard, DHCP guard, binding table, ND inspection/snooping, source guard) 57

Chapter 4 Infrastructure Services 59

Troubleshoot device management 60

Console and VTY 60
Telnet, HTTP, HTTPS, SSH, SCP, (T)FTP 60

Troubleshoot SNMP (v2c, v3) 61

Troubleshoot network problems using logging (local, syslog, debugs, conditional debugs, timestamps) 61

Troubleshoot IPv4 and IPv6 DHCP (DHCP client, IOS DHCP server, DHCP relay, DHCP options) 64

Troubleshoot network performance issues using IP SLA (jitter, tracking objects, delay, connectivity) 65

Troubleshoot NetFlow (v5, v9, flexible NetFlow) 66

Troubleshoot network problems using Cisco DNA Center assurance (connectivity, monitoring, device health, network health) 71

About the Author

Muhammad Afaq Khan started his professional career at Cisco TAC San Jose and passed his first CCIE in 2002 (#9070). He held multiple technical and management positions at Cisco San Jose HQ over his 11 years of tenure at the company before moving into cloud software and data center infrastructure IT industries.

He has worked at startups as well as Fortune 100 companies in senior leadership positions over his career. He is also a published author (Cisco Press, 2009) and holds multiple patents in the areas of networking, security, and virtualization. Currently, he is a founder at Full Stack Networker and a vocal advocate for network automation technologies and NetDevOps. He is a Cisco Certified DevNet Associate1 and was among the first 500 people #DevNet500 worldwide to pass the exam.

Preface

Congratulations! You have taken your first step towards preparing and passing the Cisco Enterprise Advanced Routing and Services (ENARSI) 300-410 V1.0 Exam.

Did you just purchase a copy? **Interested in getting access to a complimentary ENARSI Exam Quiz?** Send us an email at support@ccie in8weeks.com to get started.

This study guide is dedicated to all *those souls who will never settle for less than they can be, do, share, and give!*

What's available on the CCIEin8Weeks website

CCIEin8Weeks.com carries the supplemental resources (sold separately) that go hand in hand with this study guide to further ensure your exam success.

- Exam Prep Bundle[1] that covers all bodies of knowledge tested on the ENARSI Exam
- 4x Practice Quizzes (one for each section as per the official curriculum)
- 1x Practice Exam (to help you prepare to face the pressure of a real Cisco exam)

[1] https://bit.ly/36RWiim

Chapter 1 Layer 3 Technologies

This chapter covers the following exam topics from Cisco's official Enterprise Advanced Routing and Services (ENARSI) 300-410 V1.0 exam blueprint.

- Troubleshoot administrative distance (all routing protocols)
- Troubleshoot route map for any routing protocol (attributes, tagging, filtering)
- Troubleshoot loop prevention mechanisms (filtering, tagging, split horizon, route poisoning)
- Troubleshoot redistribution between any routing protocols or routing sources
- Troubleshoot manual and auto-summarization with any routing protocol
- Configure and verify policy-based routing
- Configure and verify VRF-Lite
- Describe Bidirectional Forwarding Detection
- Troubleshoot EIGRP (classic and named mode)
    - Address families (IPv4, IPv6)
    - Neighbor relationship and authentication
    - Loop-free path selections (RD, FD, FC, successor, feasible successor, stuck in active)
    - Stubs
    - Load balancing (equal and unequal cost)
    - Metrics
- Troubleshoot OSPF (v2/v3)
    - Address families (IPv4, IPv6)
    - Neighbor relationship and authentication
    - Network types, area types, and router types
        - Point-to-point, multipoint, broadcast, non-broadcast
        - Area type: backbone, normal, transit, stub, NSSA, totally stub
        - Internal router, backbone router, ABR, ASBR
        - Virtual link
    - Path preference
- Troubleshoot BGP (Internal and External)
    - Address families (IPv4, IPv6)
    - Neighbor relationship and authentication (next-hop, multi-hop, 4-byte AS, private AS, route refresh, synchronization, operation, peer group, states and timers)
    - Path preference (attributes and best-path)
    - Route reflector (excluding multiple route reflectors, confederations, dynamic peer)
    - Policies (inbound/outbound filtering, path manipulation)

Troubleshoot administrative distance (all routing protocols)

Administrative distance (or AD) is a number that an IP routing device uses as a tie breaker when selecting a best path, to a given destination, when two or more paths exists from two or more different routing protocols or sources. During this comparison, the lowest AD value always prevails.

AD values can range from 0 (most preferred) to 255 (least preferred). Every vendor has its own default AD values assigned to each routing protocol however they can be customized by the network engineer as desired. AD is an indication of reliability or a measure of route preference of the given routing source.

Below is a comparison of Cisco and Juniper default AD values.

Routing Source	Cisco AD	Juniper AD
Directly connected interfaces	0	0
Static route with exit interface	1	5
Static route with next-hop IP address	1	5
EIGRP summary route	5	n/a
BGP (Internal/External)	20/200	170/170
EIGRP (Internal/External)	90	n/a
OSPF (Internal/External)	110/150	10
IS-IS (L1/L2 Internal)	115/115	15/18
IS-IS (L1/L2 External)	115/115	160/165
RIP	120	100
Unknown	255	

Both Cisco and Juniper allow one-line configuration to customize a default AD value for a given routing protocol. In order to override the default AD value, you can simply go to routing process or configuration mode, and then issue "distance <new-AD>" or "set protocols <protocol> group <group-number> preference <new-AD>" on Cisco IOS and Juniper Junos OS devices respectively. Please note that there are other ways to change AD values.

For troubleshooting, it is imperative to know the main reasons behind why AD values are modified by the network engineers in the first place. There are three most common reasons that are worth noting here.

- Route redistribution
- Network migration from one routing protocol to another
- Using static route as a backup to an existing IGP route

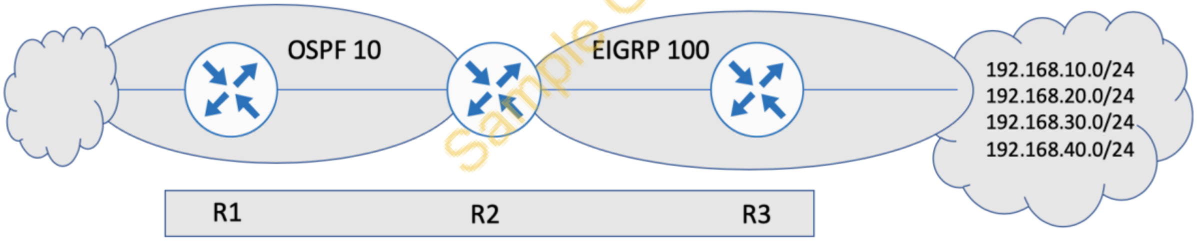
Troubleshoot route map for any routing protocol (attributes, tagging, filtering)

Route maps come in handy during IP routing redistribution as a way to select (or filter) some routes from a given routing source and redistribute them into another routing source.

There are three main components of route-map configuration in this context, i.e.

- Matching criteria defined by match statement(s), e.g. matching on an IP ACL
- Setting criteria defined by set statements(s), e.g. setting metric value, type etc.
- Application or attachment via "redistribute <routing-protocol> <#> subnets route-map <route-map-name>" command.

Let's now try this out with an example network topology. Assuming we have properly configured OSPF and EIGRP on each routing domain and now we're tasked to mutually redistributing those routes using route-maps.

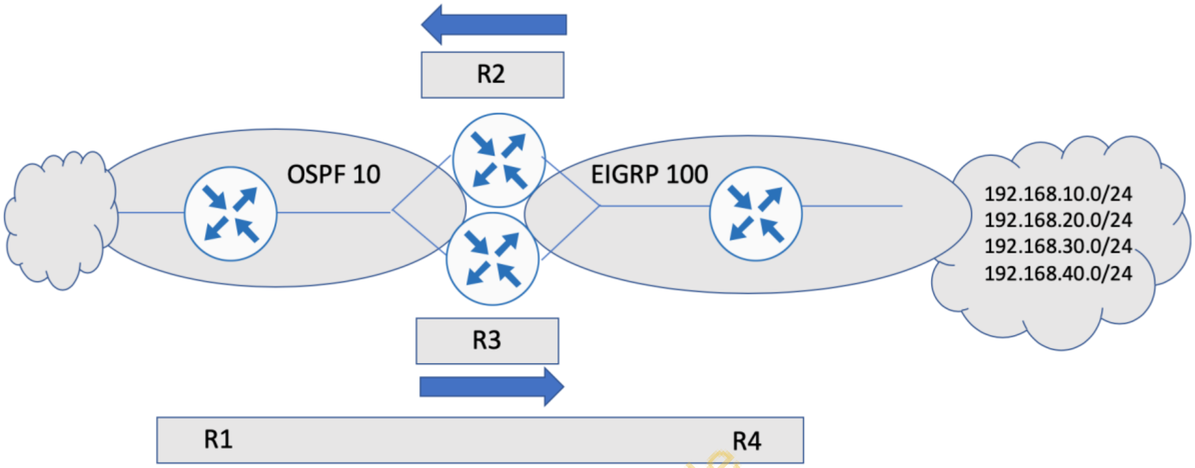


The objective is to only redistribute networks 192.168.10.0/24 and 192.168.20.0/24 into OSPF and advertise them as external type 1 (E1) routes with an external metric of 20. In order to achieve our goal, we need to configure and apply route-map on R2, the redistribution point.

```
R2(config)#access-list 10 permit 192.168.10.0 0.0.0.255
R2(config)#access-list 10 permit 192.168.20.0 0.0.0.255
R2(config)#route-map CCIEin8Weeks permit 10
R2(config-route-map)#match ip address 10
R2(config-route-map)#set metric 20
R2(config-route-map)#set metric-type type-1
R2(config)#router ospf 10
R2(config)#redistribute eigrp 100 subnets route-map CCIEin8Weeks
```

Please note that 192.168.30.0/24 and 192.168.40.0/24 will not be redistributed because of the implicit deny any at the end of the route map ACL.

In a two-way multipoint redistribution, you can use route-map tagging for route selection and filtering.



```
R2(config)#route-map EIGRP-to-OSPF deny 10
R2(config-route-map)#match tag 66

R2(config-route-map)#route-map EIGRP-to-OSPF permit 20
R2(config-route-map)#set tag 33

R2(config-route-map)#router ospf 10
R2(config-router)#redistribute eigrp 100 subnets route-map EIGRP-to-OSPF
```

Please note that you will need to apply a similar route-map configuration to R3 in order to filter OSPF to EIGRP routes.

Route maps are powerful tools when it comes to redistribution of routes. They allow network engineers to apply very fine but specific changes to routing information when it is redistributed between routing protocols. While troubleshooting route-map misconfigurations in routing applications, you're likely to find issues related to suboptimal routing or even traffic blackholing.

Troubleshoot loop prevention mechanisms (filtering, tagging, split horizon, route poisoning)

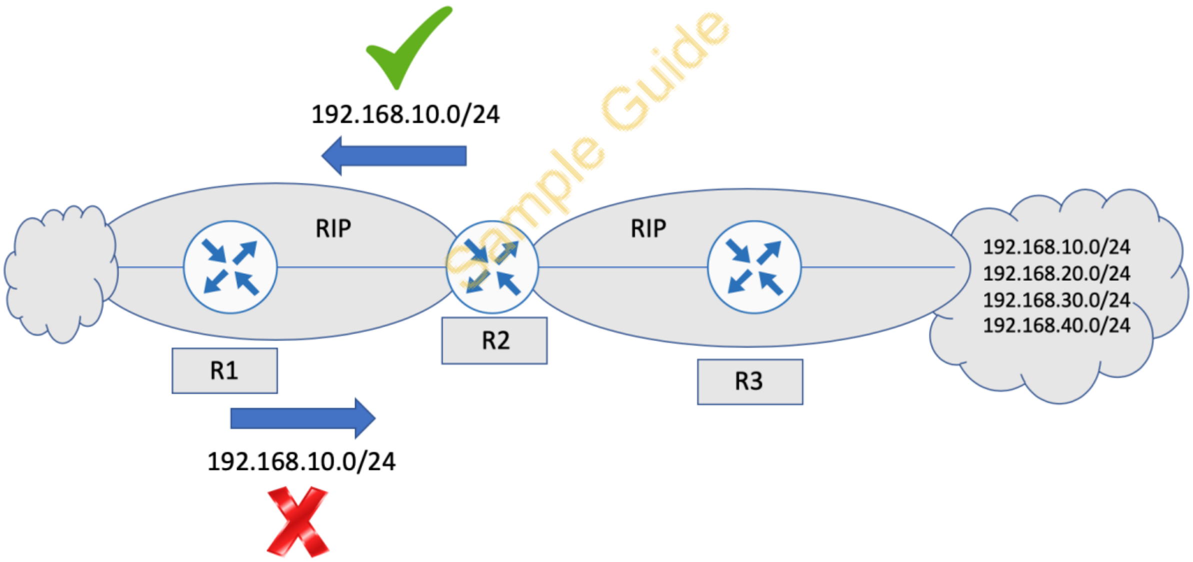
We've already discussed filtering with distribute-list as well tagging with route maps. Distance Vector Protocols (or DVRs) are based on Bellman-Ford algorithm which has several limitations.

- By default, routing loops can occur endlessly due to provision for count-to-infinity
- It doesn't scale well, so can't be used in a very large network
- Topology change is propagated slowly, i.e. router to router as opposed to flooding

In order to mitigate above issues that are inherent in DVRs, a number of features were added later on to DVRs such as RIP.

Split Horizon

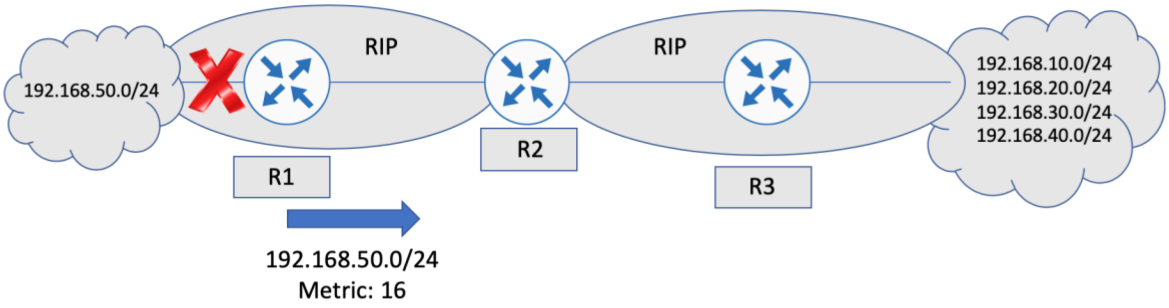
Split horizon is a mechanism to prevent routing loops in distance vector routing protocols such as RIP or IGRP. The underlying principle of operation works by never sending or advertising a route back onto the interface from which it was received or learned.



As shown in the topology above, R1 receives 192.168.10.0 from R2, but will not send the route back to R2.

Route Poisoning

Route poisoning is a method in which a router running DVR will advertise an infinity metric (16 hops in RIP) for failed routes.



Split horizon is more effective when used with route poisoning. RIP uses both split horizon with route poisoning and hold-down timers to avoid loop formation.

Let's now summarize the loop avoidance and prevention mechanisms used by various routing protocols. You can troubleshoot routing loops using the information presented in the following table.

Routing Protocol	**Loop Avoidance Feature**	**Loop Prevention Feature**
RIP v1/v2	Split Horizon Poison Reverse Triggered Updates (v2 only)	Count to Infinity Hold-down Timers
EIGRP	Split Horizon Poison Reverse	Feasibility Condition (DUAL algorithm)
OSPF intra-area	None. SPF algorithm inherently prevents loops.	
OSPF inter-area	Split Horizon (like distance vector protocols)	
IS-IS	SPF algorithm and reliable flooding	
eBGP (Inter-AS)	AS Path (attribute)	
iBGP (Intra-AS)	Split Horizon	Full-mesh requirement

Troubleshoot redistribution between any routing protocols or routing sources

In case of redistribution, your goal is to avoid both sub-optimal routing as well as AD related issues. You can avoid sub-optimal routing by eliminating multiple points of redistribution between two routing domains. On the other hand, avoiding AD related issues can be summarized into a few simple rules (not an exhaustive list).

- **Do not announce the routing information originally received from a routing source back into the same source** (e.g. RIP to IGRP, and then IGRP to RIP). You can easily filter routes using distribute-list. You can also avoid this by assigning tags to routes from a given routing source (such as RIP routes coming into EIGRP) and then filter using route-maps during redistribution (such as EIGRP routes coming back into RIP).

- **Always set a default-metric value during redistribution** (e.g. EIGRP metric) to avoid sub-optimal routing. EIGRP uses five different variables (or coefficients) for metric calculation but they are not set by default during redistribution (e.g. RIP to EIGRP) or for redistribute routes. This technique can also be used when redistribution is done between two processes of a same routing protocol (e.g. EIGRP) running on same or two different routers.
- **Avoid redistributing eBGP routes directly into an IGP**. In order to avoid this, you can simply use iBGP or another routing protocol.

Troubleshoot manual and auto-summarization with any routing protocol

Route summarization is a way to represent multiple networks with a single prefix in the form of a summary address. It comes handy in a large network where summarization can reduce both route update traffic as well as local resource consumption on the device.

There are two types of summarization techniques.

- Automatic summarization
- Manual summarization

Automatic summarization is where a router running a routing protocol that supports auto summarization feature (such as EIGRP or RIP) summarizes a set of routes on classful boundary. Auto summary feature is disabled by default with EIGRP (enabled by default with RIPv2), however it can be enabled/disabled by configuring "auto-summary" or "no auto-summary" commands respectively. Link state routing protocols such as OSPF and IS-IS do not support auto summarization.

Manual summarization is where a network engineer configures a route summary based on his/her own knowledge. The feature is available across the board with RIPv2, EIGRP, OSPF and IS-IS protocols.

Routing Protocol	Auto Summarization Support (Classful only)	Auto Summarization Default State (Classful only)	Manual/Classless Summarization Support
RIPv2	Yes	Enabled	Yes
EIGRP	Yes	Disabled	Yes

OSPF	No	n/a	Yes
IS-IS	No	n/a	Yes

When troubleshooting summarization related issues, keep the following key points in mind.

- If subnets of summary routes don't exist in the routing table, then the router will not generate a summary route (e.g. EIGRP)
- If the summary route is too broad and includes non-existent subnets, that can lead to a routing loop.

Configure and verify policy-based routing

Policy-based Routing (or PBR) is a feature which allows network engineers or admins to override the default routing policy with the help of a route map. PBR takes precedence over default destination-based routing mechanisms such as routing protocols.

To enable PBR, you need to create and apply a route-map. The route-map construct includes the matching criteria and the resulting action should successful matching occur. Route-maps are applied to router interface(s).

```
ip access-list standard A-LIST
permit 192.168.10.0 0.0.0.255

route-map CCIEin8Weeks permit 10
match ip address ACL
set ip next-hop 10.0.0.1

interface fa0/0
ip policy route-map CCIEin8Weeks
```

You can verify your route-map configuration using "show route-map" command.

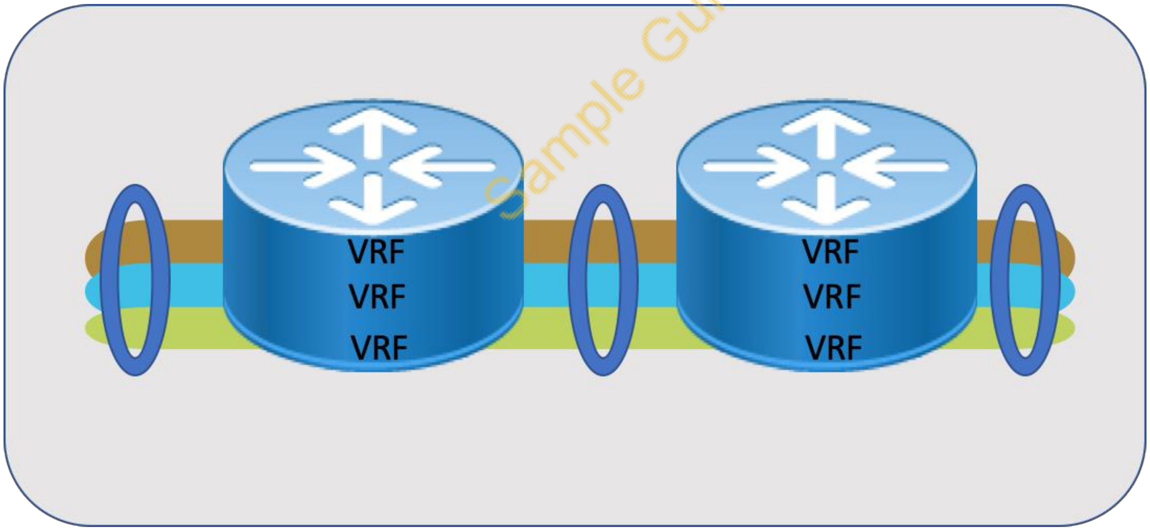
```
# sh route-map
route-map CCIEin8Weeks, permit, sequence 10
 Match clauses:
 ip address (access-lists): A-LIST
 Set clauses:
 ip next-hop 10.0.0.1
 Policy routing matches: 0 packets, 0 bytes
```

You can verify whether your route-map actually policy routing or not by sending some pings that would match your access-list and then repeating "show route-map" command.

```
# sh route-map
route-map CCIEin8Weeks, permit, sequence 10
 Match clauses:
 ip address (access-lists): A-LIST
 Set clauses:
 ip next-hop 10.0.0.1
 Policy routing matches: 6 packets, 192 bytes
```

Configure and verify VRF-Lite

Virtual Routing and Forwarding (or VRF) is feature universally available on all modern routers and L3 switches that allows multiple instances of a routing table. The VRF achieves the virtualization of the networking devices at network layer or L3. However, in order to create a VPN, you need to interconnect individual devices that are using VRFs.

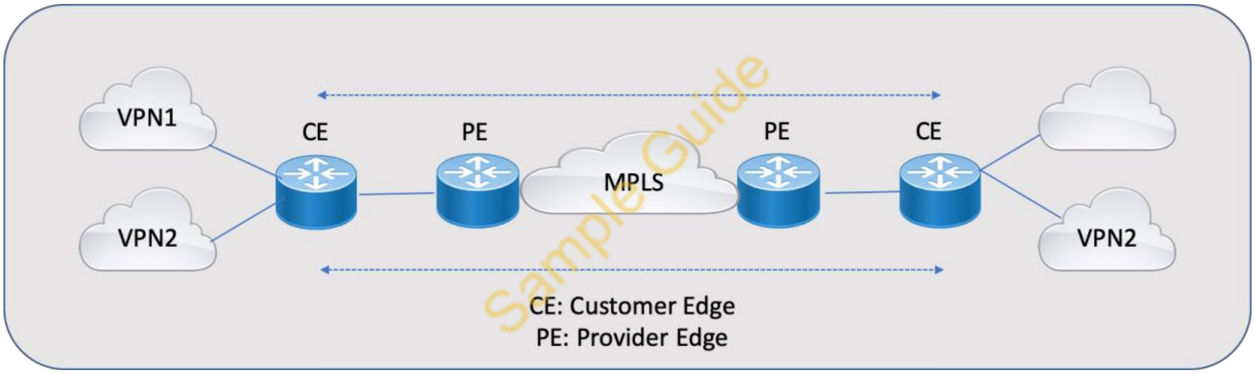


The type of data path virtualization technology to use varies depending on how far VRFs and corresponding devices are from each other. If virtualized devices are directly connected or L2 adjacent or single hop away, you have no choice but to use link or interface level virtualization e.g. using 802.1Q tag or L2 based labeling in the form of VRF to VLAN mapping.

If the virtualized devices are multiple hops away from each other, you need to use some form of tunneling to realize end to end data path virtualization. MPLS VPNs and GRE tunnels are two options that can be used to realize an end to end data path isolation. When VRFs are configured without MPLS, they are known as VRF-lite. This form of VRF configuration requires that VRF-lite interfaces are L3 or routed.

With or without MPLS, you need to understand the three device roles that the entire VRF configuration revolves around, i.e. Customer Edge (or CE), Provider Edge (or PE) and Provider (or P) devices. CE routers advertise and learn routes via CE-PE link. PE routers exchange routing information with CE routers via static routes or using a dynamic routing protocol such as BGP or RIP. PE routers only maintain VPN routes to which they are directly connected to and not all the provider's VPNs. Each VPN on PE is mapped a corresponding VRF. Finally, P routers which are also known as core routers, only reside within the provider's cloud and thus never directly attach to any of the CE routers.

With VRF-lite, multiple customers can share one CE router and one CE-PE physical link. This shared CE router keeps separate VRF tables for each customer. This is where a Multi-VRF CE device takes on some PE-like functions i.e. ability to maintain separate VRF tables.



Let's go over the day in the life of a packet for a typical VRF/VPN configuration shown above.

- CE router receives a packet, depending on the input interface the packet was received on, the router will use the corresponding routing table to look up the destination and forward it onto the PE.
- Ingress PE receives the packet from CE, it performs a VRF lookup and if the route is found, the PE router will add an MPLS label and send it into the MPLS cloud.
- Now, when egress PE receives a packet from the network, it would strip off the MPLS label but use it to identify the correct VPN/VRF routing table.
- Finally, when the destination CE router receives a packet from the PE, it uses the input interface to look up the correct VRF routing table. The packet is forwarded into the VPN if a matching route is found.

Configuring Multi-VRF CE

```
Router# configure terminal
 Enter configuration commands, one per line.  End with CNTL/Z.
 Router(config)# ip routing
 Router(config)# ip vrf v11
 Router(config-vrf)# rd 900:1
 Router(config-vrf)# route-target export 900:1
 Router(config-vrf)# route-target import 900:1
 Router(config-vrf)# exit
 Router(config)# ip vrf v12
 Router(config-vrf)# rd 900:2
 Router(config-vrf)# route-target export 900:2
 Router(config-vrf)# route-target import 900:2
 Router(config-vrf)# exit

Router(config)# interface loopback1
 Router(config-if)# ip vrf forwarding v11
 Router(config-if)# ip address 9.9.1.8 255.255.255.0
 Router(config-if)# exit

 Router(config)# interface loopback2
 Router(config-if)# ip vrf forwarding v12
 Router(config-if)# ip address 9.9.2.8 255.255.255.0
 Router(config-if)# exit

Router(config)# interface Vlan10
 Router(config-if)# ip vrf forwarding v11
 Router(config-if)# ip address 38.0.0.8 255.255.255.0
 Router(config-if)# exit
 Router(config)# interface Vlan20
 Router(config-if)# ip vrf forwarding v12
 Router(config-if)# ip address 93.0.0.8 255.255.255.0
 Router(config-if)# exit
 Router(config)# interface Vlan118
 Router(config-if)# ip vrf forwarding v12
 Router(config-if)# ip address 118.0.0.8 255.255.255.0
 Router(config-if)# exit
 Router(config)# interface Vlan208
 Router(config-if)# ip vrf forwarding v11
 Router(config-if)# ip address 208.0.0.8 255.255.255.0
 Router(config-if)# exit

Router(config)# router ospf 1 vrf vl1
 Router(config-router)# redistribute bgp 900 subnets
 Router(config-router)# network 208.0.0.0 0.0.0.255 area 0
 Router(config-router)# exit
 Router(config)# router ospf 2 vrf vl2
 Router(config-router)# redistribute bgp 900 subnets
 Router(config-router)# network 118.0.0.0 0.0.0.255 area 0
 Router(config-router)# exit

Router(config)# router bgp 900
 Router(config-router)# address-family ipv4 vrf vl2
 Router(config-router-af)# redistribute ospf 2 match internal
 Router(config-router-af)# neighbor 93.0.0.3 remote-as 100
 Router(config-router-af)# neighbor 93.0.0.3 activate
 Router(config-router-af)# network 9.9.2.0 mask 255.255.255.0
 Router(config-router-af)# exit
 Router(config-router)# address-family ipv4 vrf vl1
 Router(config-router-af)# redistribute ospf 1 match internal
 Router(config-router-af)# neighbor 38.0.0.3 remote-as 100
 Router(config-router-af)# neighbor 38.0.0.3 activate
 Router(config-router-af)# network 9.9.1.0 mask 255.255.255.0
 Router(config-router-af)# end
```

PE Configuration

```
Router# configure terminal
 Enter configuration commands, one per line.  End with CNTL/Z.
 Router(config)# ip routing
 Router(config)# ip vrf v11
 Router(config-vrf)# rd 900:1
 Router(config-vrf)# route-target export 900:1
 Router(config-vrf)# route-target import 900:1
 Router(config-vrf)# exit
 Router(config)# ip vrf v12
 Router(config-vrf)# rd 900:2
 Router(config-vrf)# route-target export 900:2
 Router(config-vrf)# route-target import 900:2
 Router(config-vrf)# exit

Router(config)# interface loopback1
 Router(config-if)# ip vrf forwarding v11
 Router(config-if)# ip address 9.9.1.8 255.255.255.0
 Router(config-if)# exit

 Router(config)# interface loopback2
 Router(config-if)# ip vrf forwarding v12
 Router(config-if)# ip address 9.9.2.8 255.255.255.0
 Router(config-if)# exit

Router(config)# interface Vlan10
 Router(config-if)# ip vrf forwarding v11
 Router(config-if)# ip address 38.0.0.8 255.255.255.0
 Router(config-if)# exit
 Router(config)# interface Vlan20
 Router(config-if)# ip vrf forwarding v12
 Router(config-if)# ip address 93.0.0.8 255.255.255.0
 Router(config-if)# exit
 Router(config)# interface Vlan118
 Router(config-if)# ip vrf forwarding v12
 Router(config-if)# ip address 118.0.0.8 255.255.255.0
 Router(config-if)# exit
 Router(config)# interface Vlan208
 Router(config-if)# ip vrf forwarding v11
 Router(config-if)# ip address 208.0.0.8 255.255.255.0
 Router(config-if)# exit

Router(config)# router ospf 1 vrf vl1
 Router(config-router)# redistribute bgp 900 subnets
 Router(config-router)# network 208.0.0.0 0.0.0.255 area 0
 Router(config-router)# exit
 Router(config)# router ospf 2 vrf vl2
 Router(config-router)# redistribute bgp 900 subnets
 Router(config-router)# network 118.0.0.0 0.0.0.255 area 0
 Router(config-router)# exit

Router(config)# router bgp 900
 Router(config-router)# address-family ipv4 vrf vl2
 Router(config-router-af)# redistribute ospf 2 match internal
 Router(config-router-af)# neighbor 93.0.0.3 remote-as 100
 Router(config-router-af)# neighbor 93.0.0.3 activate
 Router(config-router-af)# network 9.9.2.0 mask 255.255.255.0
 Router(config-router-af)# exit
 Router(config-router)# address-family ipv4 vrf vl1
 Router(config-router-af)# redistribute ospf 1 match internal
 Router(config-router-af)# neighbor 38.0.0.3 remote-as 100
 Router(config-router-af)# neighbor 38.0.0.3 activate
 Router(config-router-af)# network 9.9.1.0 mask 255.255.255.0
 Router(config-router-af)# end
```

Verifying VRF Configuration

There are plenty of Cisco IOS CLIs that you can use to verify an active VRF configuration.

Show ip protocols vrf <vrf-name> displays routing protocol information related to a VRF.

Show ip route vrf <vrf-name> displays IP routing table information related to a VRF.

Describe Bidirectional Forwarding Detection (BFD)

Bidirectional Forwarding Detection (or BFD) is a forwarding path failure detection method between two IP devices and can be configured for interfaces, data links and forwarding planes.

BFD (RFC 5880) uses a low overhead detection protocol that can be enabled on per-interface or per-protocol basis. BFD protocol uses a three-way handshake to establish and tear down a session between the two pre-configured devices. You can optionally configure authentication using either a simple cleartext password or MD5/SHA-1.

Protocols such as OSPF, IS-IS, BGP or even RIP can be used to start a BFD session, thus allowing these protocols to use BFD for failure detection as opposed to using their built-in keepalive based mechanisms.

BFD session can operate in asynchronous or demand modes. In asynchronous mode, both endpoints or devices send periodic hello messages to each other, and if a given number of those messages go unacknowledged, BFD will mark the given session as Down.