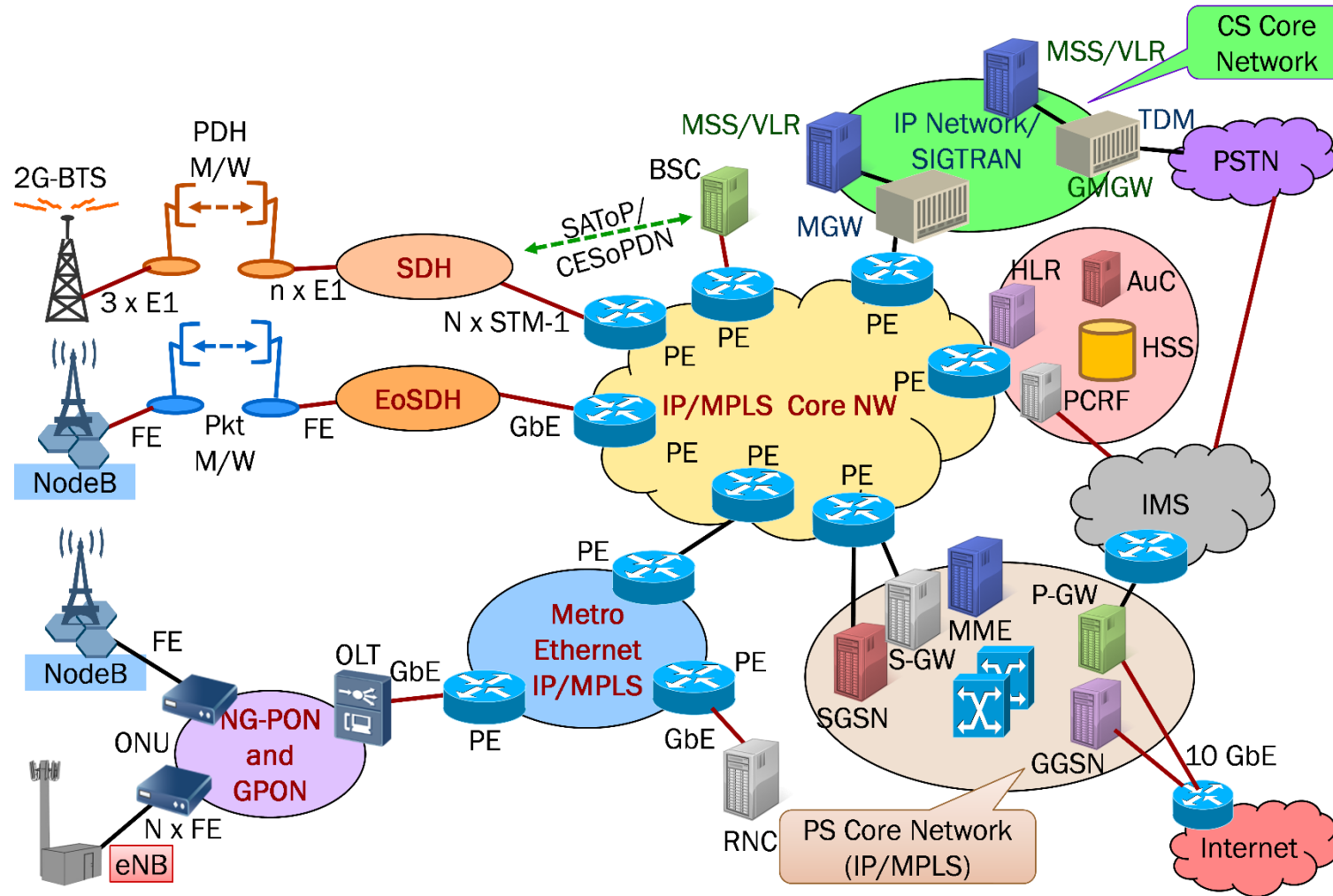
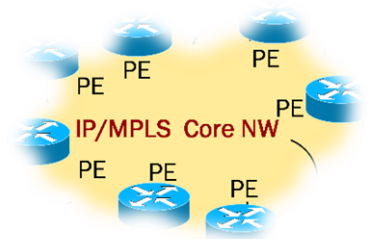


# Servicios MPLS Layer 3 - VPRN



# Virtual Private Routed Networks (VPRN)

## Significado y motivaciones



# Virtual Private Routed Networks:

---

- Está definido por la **RFC 4364**.
- Provee un servicio:
  - **ruteado**
  - **multipunto** a los clientes sobre una red IP/MPLS.

# Virtual Private Routed Networks:

---

- El servicio VPRN es **similar a muchos aspectos** a los servicios de **Layer 2 VPN**:
  - Los **datos** de los clientes son:
    - **Transportados** a través de la red del proveedor de servicios **encapsulados**.
  - Con un **label stack** de dos etiquetas MPLS,
    - la inner para el **service label**
    - la outer para el **transport label**.

## Virtual Private Routed Networks:

---

- La red de core del proveedor de servicios es transparente para el cliente.
- El proveedor de servicios puede utilizar la misma infraestructura de red IP/MPLS
  - Para soportar varios servicios diferentes VPRN
  - Como así también servicios Layer 2 como ser VPWS o VPLS.

## VPRN – Diferencias servicios Layer 2 y servicios Layer 3:

---

- Existen también algunas diferencias entre estos servicios Layer 2 y este servicio Layer 3:
  - Los routers CE realizan un peering con los routers PE utilizando algún protocolo de enrutamiento como ser RIP, OSPF o BGP.
  - Los routers PE mantienen un router privado y virtual para cada servicio VPRN configurado.

## VPRN – Diferencias servicios Layer 2 y servicios Layer 3:

---

- Existen también algunas diferencias entre estos servicios Layer 2 y este servicio Layer 3:
  - Multiprotocol BGP (MP-BGP) es utilizado para transportar la información de rutas de los clientes a través de la red del proveedor de servicios.
  - MP-BGP también es utilizado para señalar los services labels entre los routers PE de ingreso y de egreso, en lugar de Targeted LDP.

- Desde la **perspectiva del cliente**:
  - Muestra a los clientes como si todos los sitios estuvieran **conectados a un router IP**.
  - Puede operar sobre un único site local, como **sobre múltiples y sitios geográficamente dispersos**.
  - **Distribuye las rutas del cliente** entre los diferentes sitios del cliente de manera que el tráfico sea forwardado correctamente a través de la red.
  - Existe una **redundancia** basada en la red del proveedor de servicio.



## VPRN – Ventajas del servicio:

---

- Desde la **perspectiva del proveedor del servicio:**
  - Solo los routers PE requieren configuración adicional para brindar el servicio VPRN.
  - Existe una clara demarcación de funcionalidad entre las redes del cliente y del proveedor.
  - Las redes de los clientes son separadas y espacios de direcciones solapados entre clientes pueden ser implementados.
  - El proveedor de servicio puede aplicar QoS, traffic shapping o policy **específicos cada servicio VPRN.**

### Router CE:

- Solo **updates IPv4 como así datos**, son forwardados hacia el router PE.
- Este router CE **no necesita tener ninguna configuración específica** que lo haga parte del dominio MPLS VPN.
- El único requerimiento en el router CE es que un **protocolo de routing, estático o dinámico**, sea habilitado en el router para intercambiar información de routing IPv4 con el router PE al cual está conectado.

### Router PE,

- Este debe ser capaz de aislar el tráfico de los clientes si más de un cliente está conectado al router PE.
- De esta forma, cada cliente se asigna a una tabla de ruteo independiente como si estuvieran conectados a routers PE individuales.
- El routing a través del backbone del SP se realiza utilizando un proceso de routing en base a la tabla de ruteo global.

### Router P,

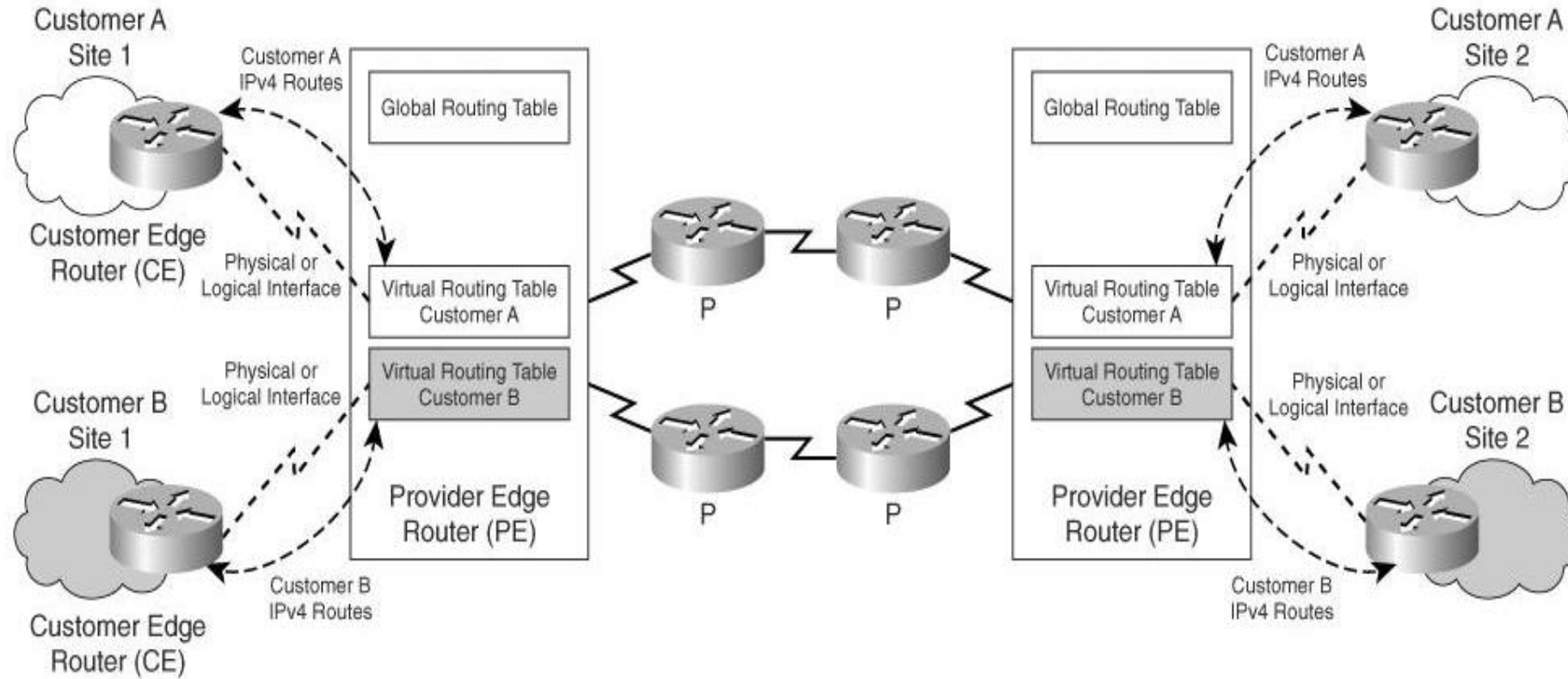
- Provee el label switching entre routers PE.
- Solo son responsables de este switching de paquetes.
- Desconocen las rutas VPN por lo que no participan del proceso de routing MPLS VPN.

## VPRN – Roles de los routers en la red:

---

- Los routers CE en el cliente:
  - desconocen la implementación de los routers P
  - la topología interna de la red del SP es transparente para el cliente.
- Los routers PE intercambian rutas IPv4 con el router CE al que están conectado

# VPRN – Roles de los routers en la red:

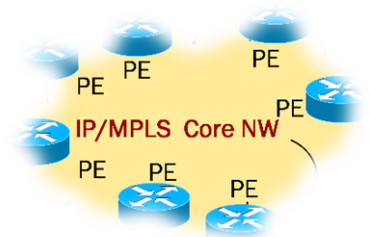


## VPRN – Componentes de la red:

---

- El plano de datos de un servicio VPRN es muy similar al de los servicios de Layer 2.
- Existen una serie de **nuevos componentes** introducidos en los servicios VPRN:
  - VRF
  - Route Distinguisher(RD)
  - Route Target (RT)
  - MP-BGP

# Componentes de la red





# Virtual Routing and Forwarding (VRF):

---

- Para cada VPN en la red del proveedor,
  - Necesitamos tener en cada router PE una instancia de routing privado, separado e independiente el uno del otro.
  - Necesitamos que cada VPN tenga su propia tabla de ruteo.
- Esta tabla de ruteo privada se denomina VRF routing table.
- La interfaz en un router PE hacia un router CE puede pertenecer a una única VRF.
- En función de esto, todos los paquetes recibidos en esa interfaz pertenecerán a la VRF en cuestión.

## Virtual Routing and Forwarding (VRF):

---

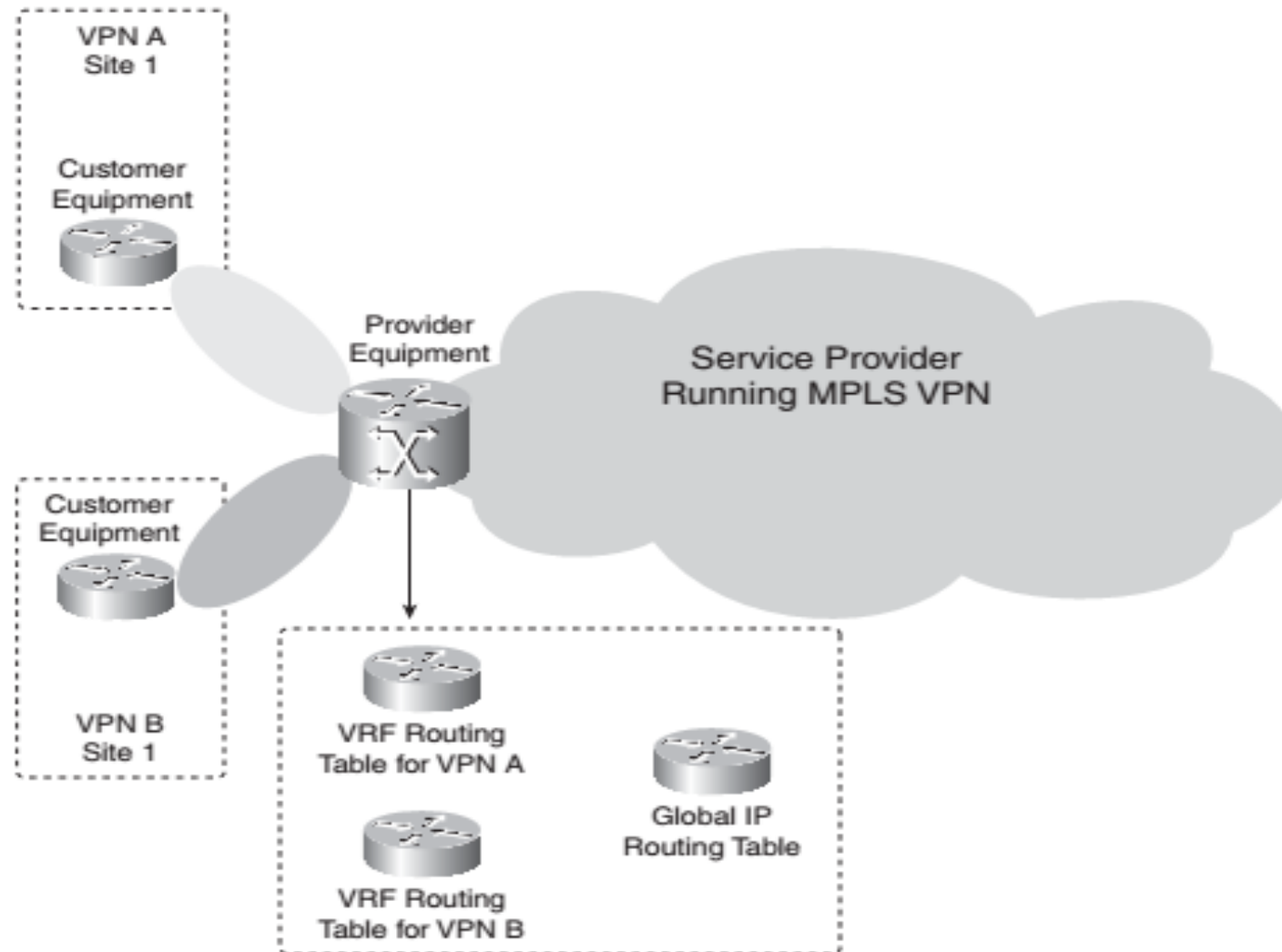
- El comando para crear las VRF en un router PE es “ip vrf”.
- El comando utilizado para asignar interfaces PE-CE en el router PE a una VRF es “ip vrf forwarding”.
- Es posible asignar una interfaz a un único VRF.
- Pero es posible asignar varias interfaces diferentes a un mismo VRF.

## Virtual Routing and Forwarding (VRF):

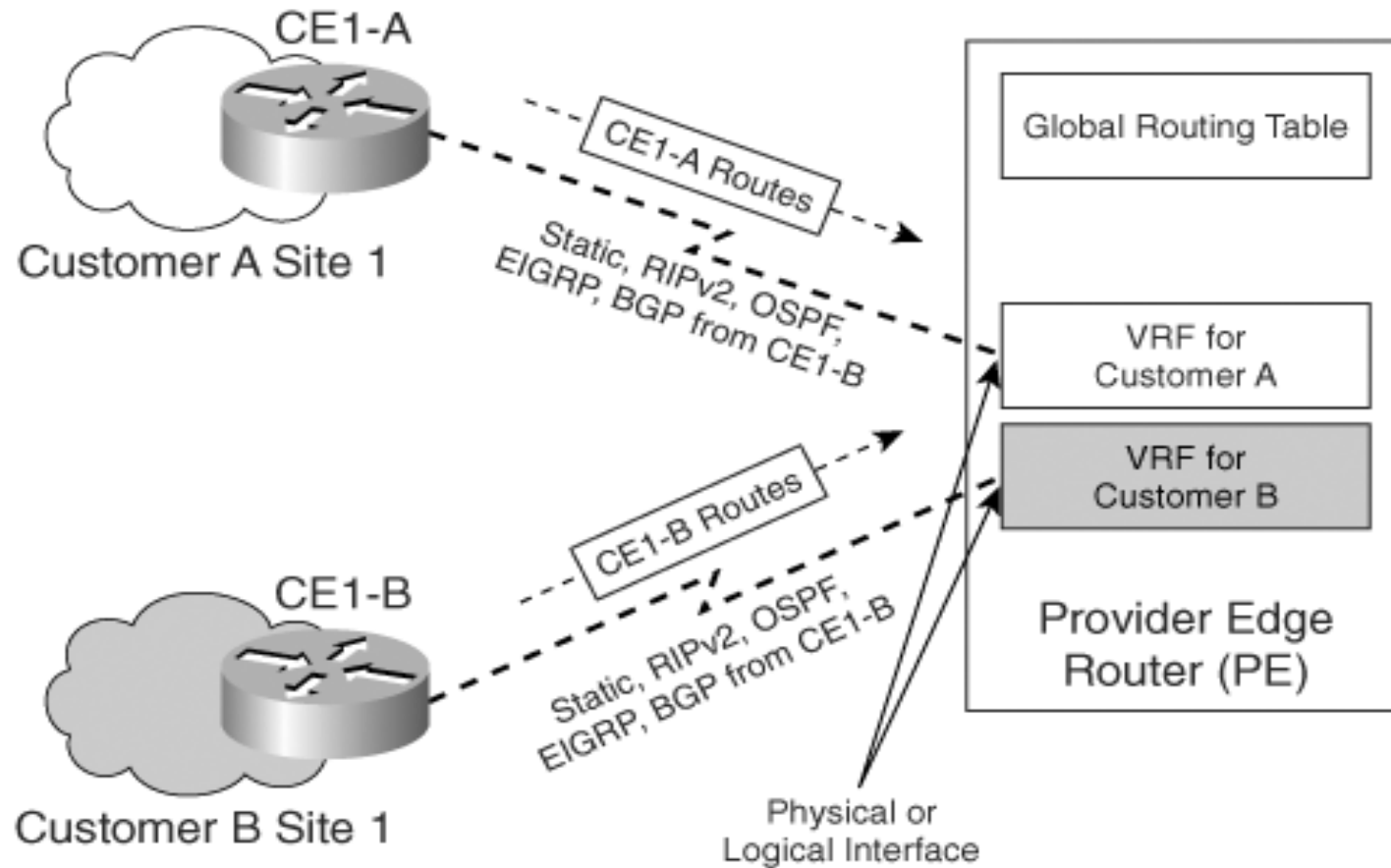
---

- El router PE **creará automáticamente la VRF routing table**.
- Esta tabla **no tendrá diferencia alguna con una tabla de ruteo “regular”** o global o “default routing table”.
- Las interfaces que no son asignadas específicamente a ningún VRF **formarán parte de la tabla de ruteo global del router**.

# Virtual Routing and Forwarding (VRF):



# Virtual Routing and Forwarding (VRF):



## Virtual Routing and Forwarding (VRF):

---

- Es una instancia virtual de ruteo.
- Separada de la instancia global de ruteo en el dispositivo.
- Permite que **múltiples instancias de una tabla de ruteo coexistan** dentro de un mismo router al mismo tiempo.
- Entre otros beneficios, tener una VRF nos permite la **duplicación (solapamiento) de direcciones IP entre diferentes clientes**, ya que las rutas se procesan basadas en las etiquetas.

## Virtual Routing and Forwarding (VRF):

---

- Cuando activamos una nueva instancia o VRF:

el router **puede y debe ejecutar** sus propios procesos de routing como ser BGP, OSPF, ISIS o static.

- También debemos indicar en cada interface a que VRF queremos que pertenezca.
- Podemos decir que:

*los VRF son a la capa 3, lo que las vlans son a la capa 2.*

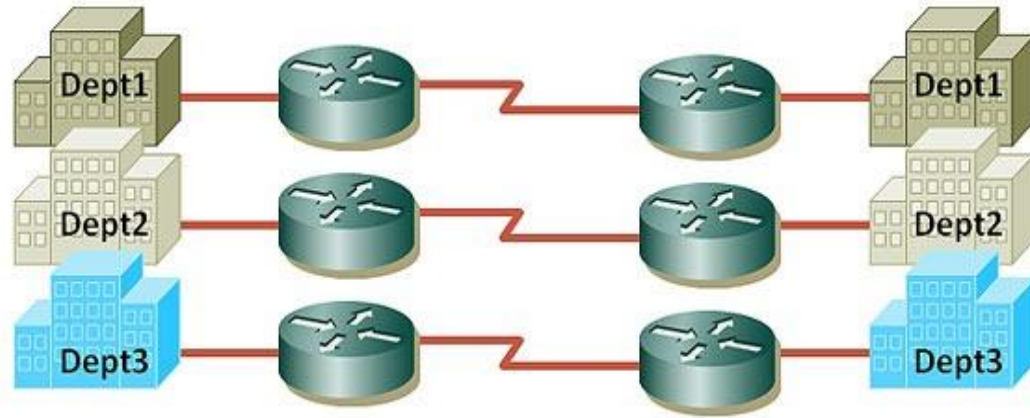
## Virtual Routing and Forwarding (VRF):

---

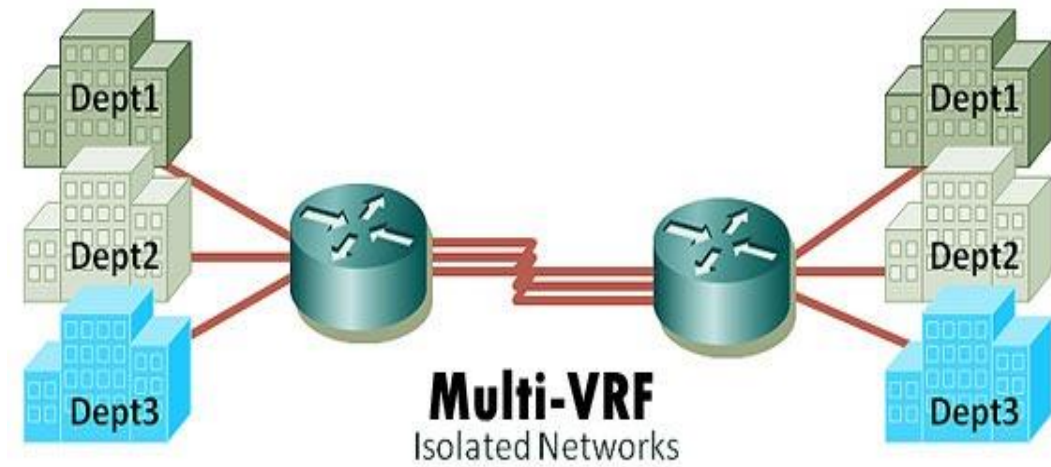
- A menos que se hayan aplicado políticas de importación/exportación, **las rutas** (y por ende los paquetes) **no podrán pasar de un VRF a otro**
- Esto de manera similar a lo que ocurre en una VLAN.
- Es decir, los paquetes que ingresan en un router en el VRF A solo podrán alcanzar rutas de la tabla de ruteo de este VRF A.
- En sintonía a las VLANs, las **VRF solo tienen** **significancia local en el router.**



# Virtual Routing and Forwarding (VRF):



Multiple Routers  
Isolated Networks



## Virtual Routing and Forwarding (VRF):

---

- Las VRFs no son una característica propia de MPLS.
- La **implementación más sencilla de una VRF es la del tipo VRF lite.**
  - Cada router dentro de la red participa dentro del entorno de ruteo virtual de una manera basada en pares.
  - Es sencilla de desplegar y apropiada para pequeñas y medianas redes.
  - No son escalables, ya que se requieren que sean implementadas en cada router, incluidos los intermedios en la red.

## Virtual Routing and Forwarding (VRF):

---

- Para el caso de una red extensa, existe las **full VRF**, o **VRF implementadas en MPLS L3 VPN**.
  - Solo debemos configurar o terminar las VRFs en los routers PE.
  - El core de la red no necesita conocer sobre las VRF y esto la hace escalable y altamente extensibles.
- Pero como entonces interconectamos estas VRFs?
  - Para esto se desarrolló **Multiprotocol BGP (MP-BGP or BGP-MP)** que se utiliza únicamente para intercambiar rutas entre diferentes VRFs.

## Virtual Routing and Forwarding (VRF):

---

- Estas rutas son importadas dentro de BGP con un “tag” propio para el intercambio.
- Es importante destacar que los routers de core MPLS pueden reenviar el tráfico desconociendo la configuración de las VRF.
- Estos tags que utilizaremos para importar o exportar rutas se denominan “route targets”.

## RD (Route Distinguisher) y RT (Route Target):

---

- MPLS nos da la **posibilidad de utilizar el mismo backbone** para distintos clientes.
  - Manteniendo una separación entre ellos.
- Esto permite, por ejemplo, **tener diferentes clientes utilizando el mismo rango de direcciones IPv4 privadas.**
- ¿Cómo puede un proveedor de servicio ofrecer su infraestructura para conectividad a distintos sitios remotos de un cliente A y un cliente B **sin mezclar su información de enrutamiento?**

## RD (Route Distinguisher) y RT (Route Target):

---

- Como anteriormente mencionamos, los VRF permiten reutilizar el espacio de direccionamiento IP privado en forma aisladas entre diferentes dominios de routing.

Por ejemplo, podríamos tener una red en la que conectamos tres sitios de clientes, en los que en cada uno está configurada la subred 192.168.0.0/24.

- Para esto, le asignamos a cada cliente su propia VRF
- De manera que esa superposición de subredes quede aislada dentro de su respectivo dominio de routing.

## RD (Route Distinguisher) y RT (Route Target):

---

- Esto funciona correctamente dentro de un mismo router.
- Pero cuando interconectamos varios routers con varios VRF dentro de una red?
  - Necesitamos de alguna forma poder administrar o tener un seguimiento sobre a qué cliente pertenece cada paquete perteneciente a la misma subred 192.168.0.0/24.
- *Para esto utilizamos los route distinguishers.*

## RD (Route Distinguisher):

---

- Estos **distinguen un set de rutas (de un VRF) de otro.**
- Es un **número único**
- **Se antepone a cada ruta** dentro de un VRF
  - Esto para identificarla como perteneciente a un VRF o cliente particular.
- Cada instancia VRF en un router PE tiene asignado un RD.



## RD (Route Distinguisher):

---

- El prefijo derivado de la combinación del prefijo IPv4 y el RD:
  - es denominado **vpn4 prefix**
- Estos vpn4 prefix son transportados con la ruta a lo largo vía MP-BGP cuando se intercambian rutas VPN con otros routers PE.

## RD (Route Distinguisher):

---

- El RD tiene **64 bits de longitud** que comprenden tres campos:
  - el campo tipo (dos bytes),
  - El campo administrador
  - El campo valor.
- Existen actualmente definidos tres formatos de RD que pueden ser utilizados

Type 0	2-byte ASN	4-byte value
Type 1	4-byte IP	2-byte value
Type 2	4-byte ASN	2-byte value

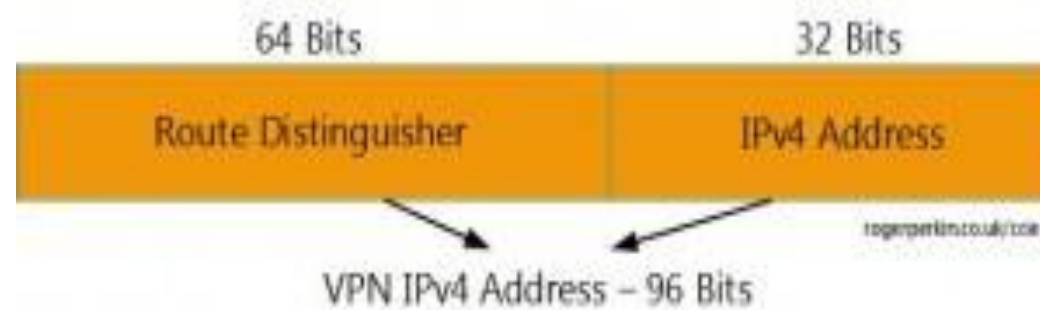
## RD (Route Distinguisher):

---

- Utilizando el Route Distinguisher, entonces:

**transformamos la dirección IP de 32 bits en una dirección de 96 bits, única en la red.**

- De esta forma los routers Provider Edge (PEs) en MPLS no anuncian redes de 32 bits vía mBGP, sino un prefijo de 96 bits.



## RD (Route Distinguisher):

---

- El **modelo más utilizado** es el formato ASN:nn donde ASN es el número de sistema autónomo provisto por el IANA y nn es un número que el service provider asigna unívocamente al VRF
- Por ejemplo:
  - Una ruta para la subred 192.168.2.0/24 en el VRF del Site\_B
  - Finalmente se publica como 65535:20:192.168.2.0/24,
    - donde 65535 es el número de AS
    - el 20 es el site ID.

# RD (Route Distinguisher):

---

- Como se configura?

```
sydney#conf t
```

```
sydney(config)#ip vrf ?
```

WORD VPN Routing/Forwarding instance name

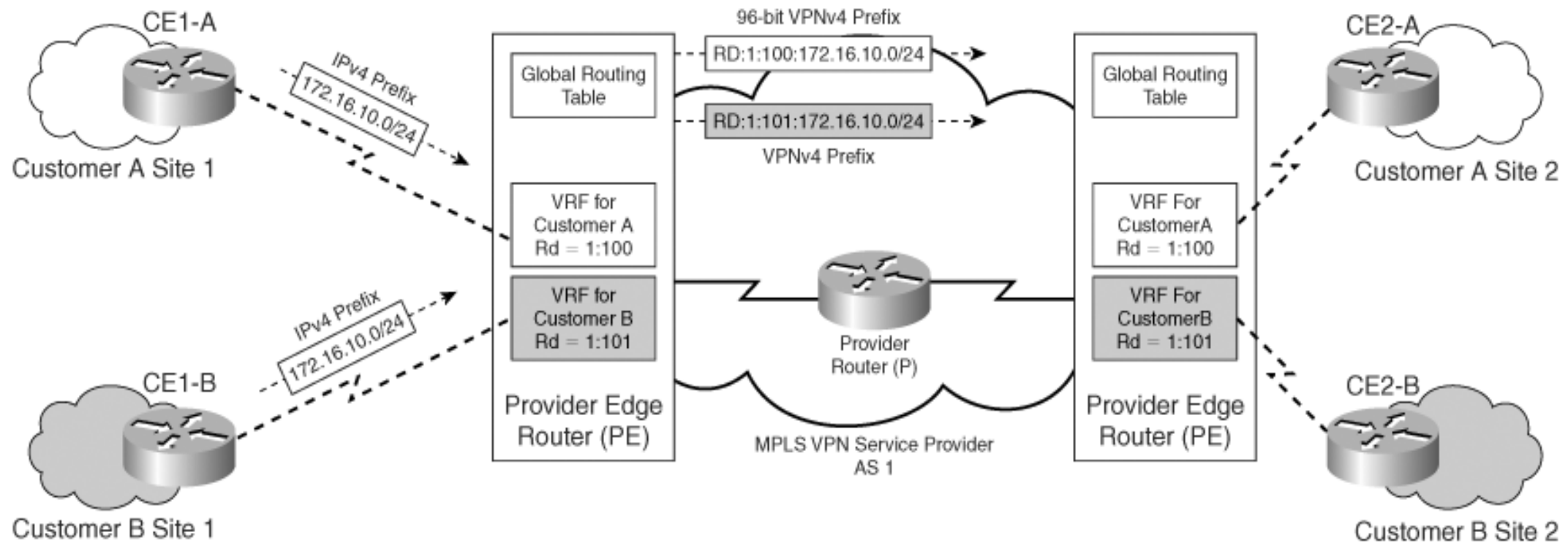
```
sydney(config)#ip vrf cust-one
```

```
sydney(config-vrf)#rd ?
```

ASN:nn or IP-address:nn VPN Route Distinguisher

```
sydney(config-vrf)#rd 1:1
```

# RD (Route Distinguisher):



## RT (Route-Target):

---

- Los **RD** se utilizan **para indicar la pertenencia a una VPN o VRF**.
- Que sucede con la **comunicación entre sitios pertenecientes a diferentes VPN o VRF?**.
- **NO tendrán la posibilidad** de conectarse con un sitio de otro cliente debido a que no coincidirían los RDs.
- El concepto de tener sitios del Cliente A pudiendo ser capaces de conectarse con sitios del Cliente B es denominado **extranet VPN**.
- El caso simple de comunicación únicamente entre sitios del mismo VPN o VRF es denominado **intranet VPN**.

## RT (Route-Target):

---

- La comunicación entre sitios es controlada por otra feature MPLS VPN denominada RTs.
- Un RT o Route-Target es una **comunidad extendida de BGP**

**Indica que rutas deben ser importadas de MP-BGP en un VRF.**

- **Exportar un RT significa** que las rutas vpnv4 exportadas reciben una BGP extended community adicional (el RT)
- Esto según configuración bajo ip vrf en los routers PE, cuando la ruta es redistribuida desde la tabla de ruteo de un VRF hacia MP-BGP.



## RT (Route-Target):

---

- **Importar un RT significa** que las rutas vpnv4 recibidas desde MP-BGP son chequeadas en búsqueda de coincidencia de la extended community (RT) con alguna configurada.
- Si el resultado es una coincidencia, el prefijo es colocado en la tabla de ruteo del VRF como una ruta IPv4. Si no se produce coincidencia, el prefijo es rechazado.

## RT (Route-Target):

---

- El comando para configurar RTs para un VRF es:
- `route-target {import | export | both} route-target-extcommunity.`

## RT (Route-Target):

---

- Es un campo de **8 bytes (64 bits)** que es parte de los BGP extended Communities Attribute definidos en la RFC 4360.
- Mediante este campo, *definimos que prefijos serán importados y exportados en los routers PE.*
- Cuando exportamos prefijos en un VRF, agregamos a estos una comunidad de Route-Target.
- De esta forma cuando el PE en un sitio remoto tiene que importar prefijos en el VRF, puede identificarlos fácilmente, escogiendo que prefijos quiere importar basándose en este valor.

## RT (Route-Target):

---

- El route target tiene formato similar al route distinguisher, pero sirve para propósitos diferentes.
- Que el route target pueda tener el mismo formato que el route Distinguisher, no significa que exista una relación directa entre estos dos.
- Mientras que:
  - los RD son utilizados para mantener la identificación univoca de rutas similares en diferentes VRFs.
  - los RT pueden ser utilizados para compartir rutas entre estos VRFs, es decir podemos utilizar los RT para controlar que rutas importamos y exportamos entre los VRFs.

## RT (Route-Target):

---

- Imaginemos que:
  - Utilizando el mismo RD para todos los sitios del cliente A,
  - No deseamos que todas las sedes se puedan conectar a todas las sedes.
  - Podríamos tener una configuración en donde el Site-2 tenga visibilidad de los prefijos de Site-1 pero no para Site-3.
  - Igualmente para Site-3, podríamos hacer que tuviera visibilidad de Site-1 pero no de Site-2.

## RT (Route-Target):

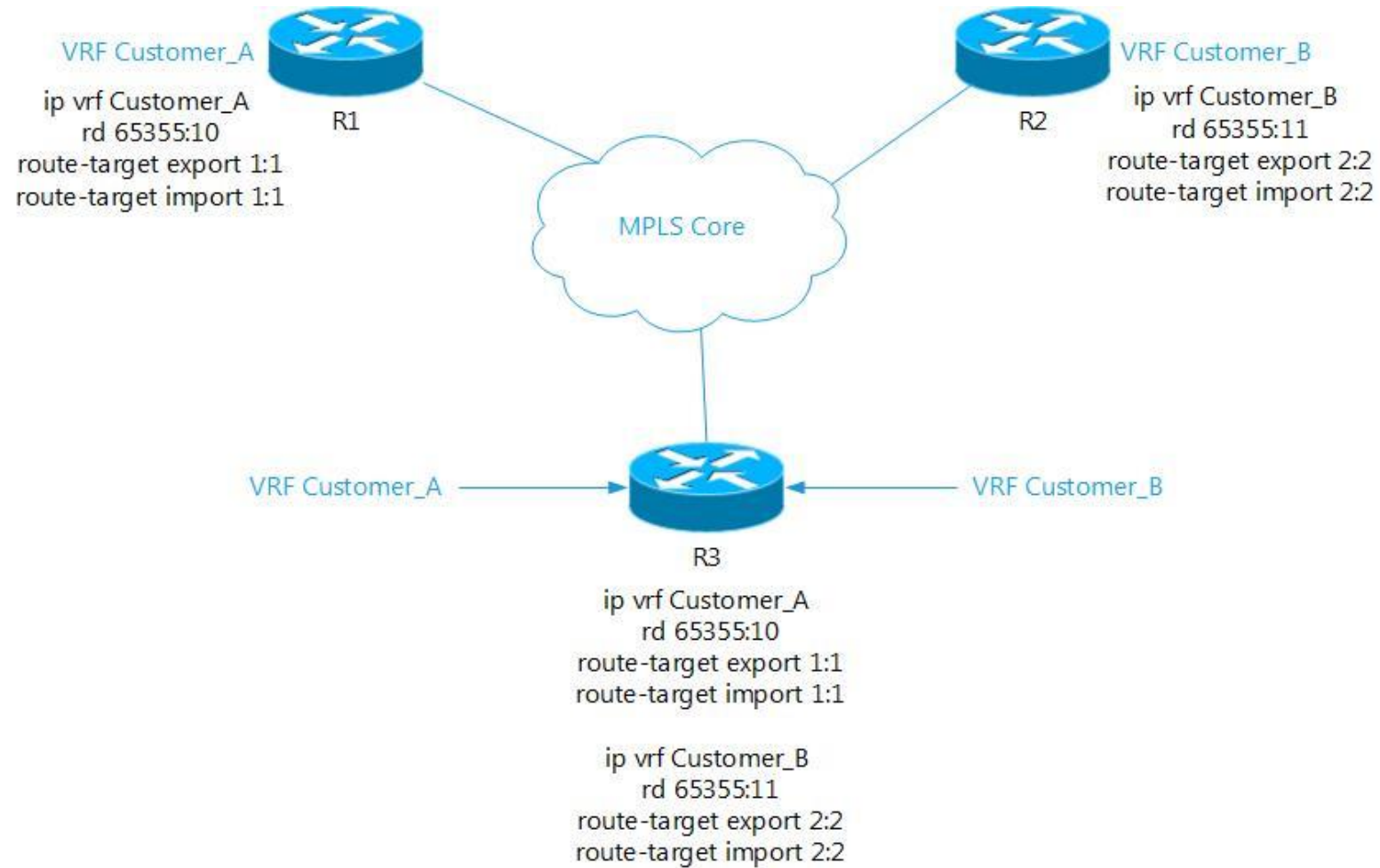
---

- Ejemplo:

- El R3 tiene 2 VRFs configurados, el vrf Customer\_A y el vrf Customer\_B
- En el router R3 queremos exportar e importar prefijos de los vrf Customer\_A y Customer\_B.
- Sin embargo, en el R1 solamente queremos importar y exportar prefijos para el vrf Customer\_A y en el R2 solamente queremos importar y exportar prefijos para el vrf Customer\_B.
- De esta forma, la configuración seria la siguiente:

# RT (Route-Target):

- Ejemplo:



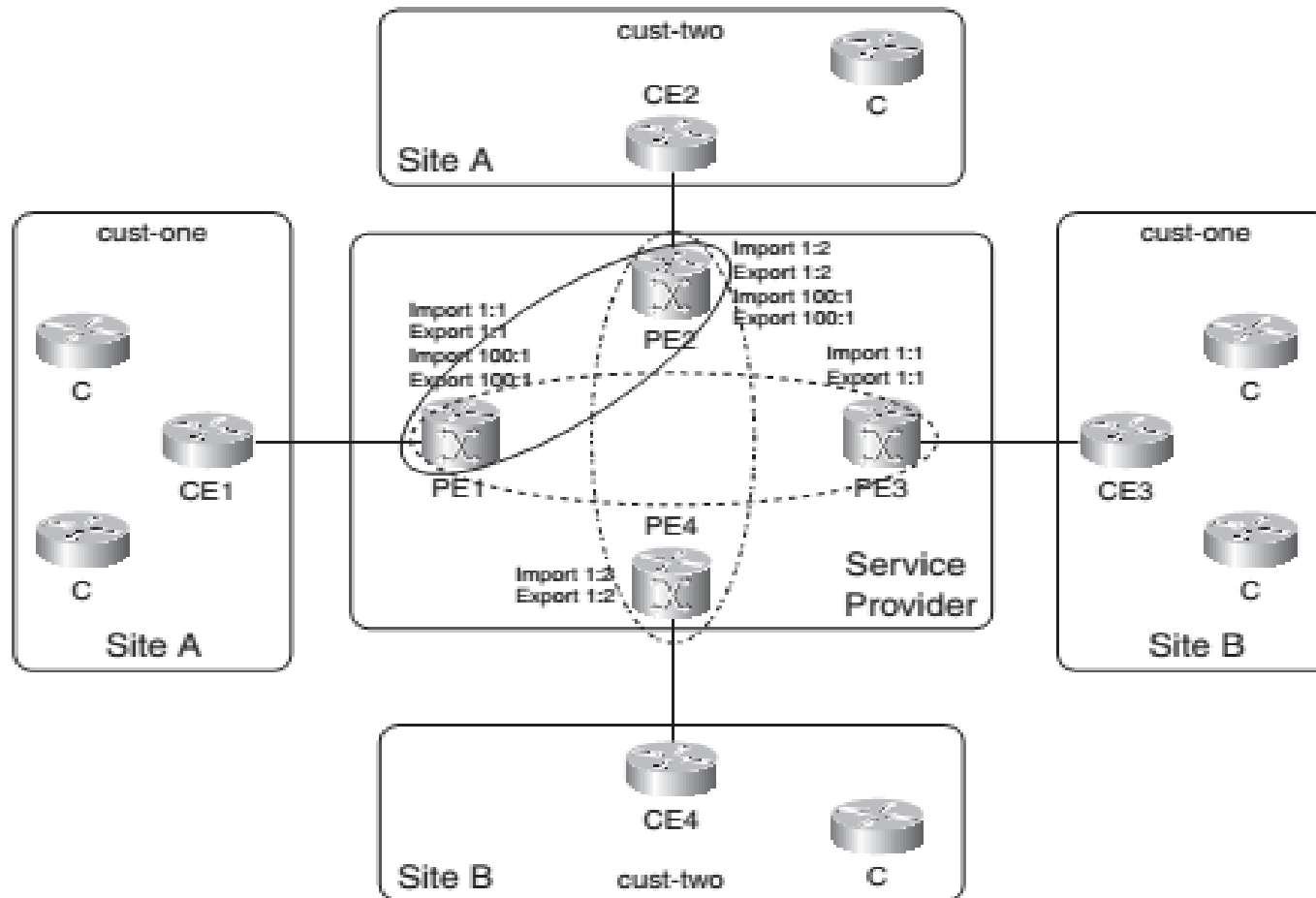
## RT (Route-Target):

---

- Supongamos el siguiente ejemplo:
  - El Site A y el Site B del VRF cust-one deben poder comunicarse entre si.
  - Lo mismo para el Site A y el Site B del VRF cust-two.
- El RT que el VPN cust-one utiliza es 1:1.
- El RT que el VPN cust-two utiliza es 1:2.
- Ahora supongamos que el el Site A del VRF cust-one necesita conectarse con el Site A del VRF cust-two.
- El RT 100:1 es importado y exportado por el Site A del VRF cust-one y el Site A del VRF cust-two en el PE1 y el PE2 para lograr esta conexión del tipo extranet.



## RT (Route-Target):



# RT (Route-Target):

---

PE1:

!

```
ip vrf cust-one
```

```
rd 1:1
```

```
route-target export 1:1
```

```
route-target export 100:1
```

```
route-target import 1:1
```

```
route-target import 100:1
```

!

PE2

```
ip vrf cust-two
```

```
rd 1:2
```

```
route-target export 1:2
```

```
route-target export 100:1
```

```
route-target import 1:2
```

```
route-target import 100:1
```

!

## Multi Protocol - BGP:

---

- BGP-4 esta descripto en el RFC 1771.
- Este RFC solo describe el uso de BGP para transportar prefijos IPv4.
- El RFC 2858 “Multiprotocol Extensions for BGP-4” fue descripto como una extensión de BGP .
- Este permite transportar otra información de ruteo que tan solo IPv4.

## Multi Protocol - BGP:

---

- Un BGP speaker le hace conocer a sus peer que soporta extensiones multiprotocolos para BGP-4.
- Esto utilizando una serie de **capacidades de publicación**.
- Cada peer BGP envía a su extremo las capacidades que este soporta.
- Las capacidades que los peers tienen en común, entonces pueden ser utilizadas.

## Multi Protocol - BGP:

---

- Para soportar el funcionamiento Multiprotocolo (MP) de BGP en Cisco, el proceso de routing BGP tiene el concepto de **address family**.
- Cuatro address family actualmente están soportadas:
  - IPv4,
  - IPv6
  - vpnv4
  - vpnv6.
- Utilizaremos el **address family vpnv4** bajo el proceso bgp para configurar las sesiones BGP vpnv4 y los parámetros que los routers PE necesitan.

## Multi Protocol - BGP:

---

- El **draft ietf-idr-bgp-ext-communities** define los “**extended community attribute**”.
- Muchos BGP “extended community attributes” son definidos, pero solo uno es requerido para MPLS VPN:

- el RT extended community.**

- Como dijimos anteriormente este:

*“indica a los routers PE que tiene configurado BGP que rutas deben ser importadas en un VRF”*

# Multi Protocol - BGP:

---

- Supongamos:

- la ruta vpnv4:

- 1:1:10.10.100.1/32**

- con RTs: 1:1,

- 100:100

- 100:101.

## Multi Protocol - BGP:

---

- Solo los VRF que tiene configurados para importar al menos uno de estos RTs

insertan la route IPv4 10.10.100.1/32 en la tabla de routing VRF.

- Respecto a la configuración de BGP:

- Primero necesitamos definir el vecino BGP en la configuración global de BGP.

- Luego, necesitamos habilitar el vecino BGP en la vpnv4 address family especificando el keyword “actíivate”.

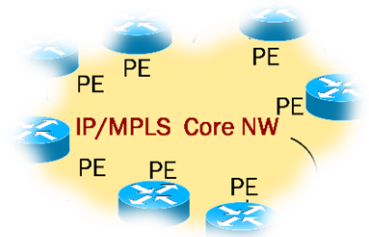


# Multi Protocol - BGP:

---

```
router bgp 1
neighbor 10.200.254.2 remote-as 1
neighbor 10.200.254.2 update-source Loopback0
!
address-family ipv4
redistribute rip
neighbor 10.200.254.2 activate
exit-address-family
!
address-family vpnv4
neighbor 10.200.254.2 activate
neighbor 10.200.254.2 send-community both
exit-address-family
!
```

# Propagación de rutas



## Propagación de rutas:

---

- El router PE recibe rutas IPv4 desde el router CE.
  - Esto a través de un IGP o un external BGP (eBGP).
- Estas rutas IPv4 son introducidas en la tabla de ruteo del VRF correspondiente.
- En que VRF son introducidas las rutas depende la configuración de la interfaz del router PE que lo conecta hacia el router CE

## Propagación de rutas:

---

- Estas rutas son **adjuntadas con el RD que es asignado al VRF** correspondiente.
  - Esto las convierte entonces en *prefijos vpnv4*.
- Estos prefijos luego son **puestos dentro del MP-BGP**.
- **BGP se encarga de distribuir estas vpnv4 routes** o prefix a todos los routers PE dentro de la red MPLS VPN.

## Propagación de rutas:

---

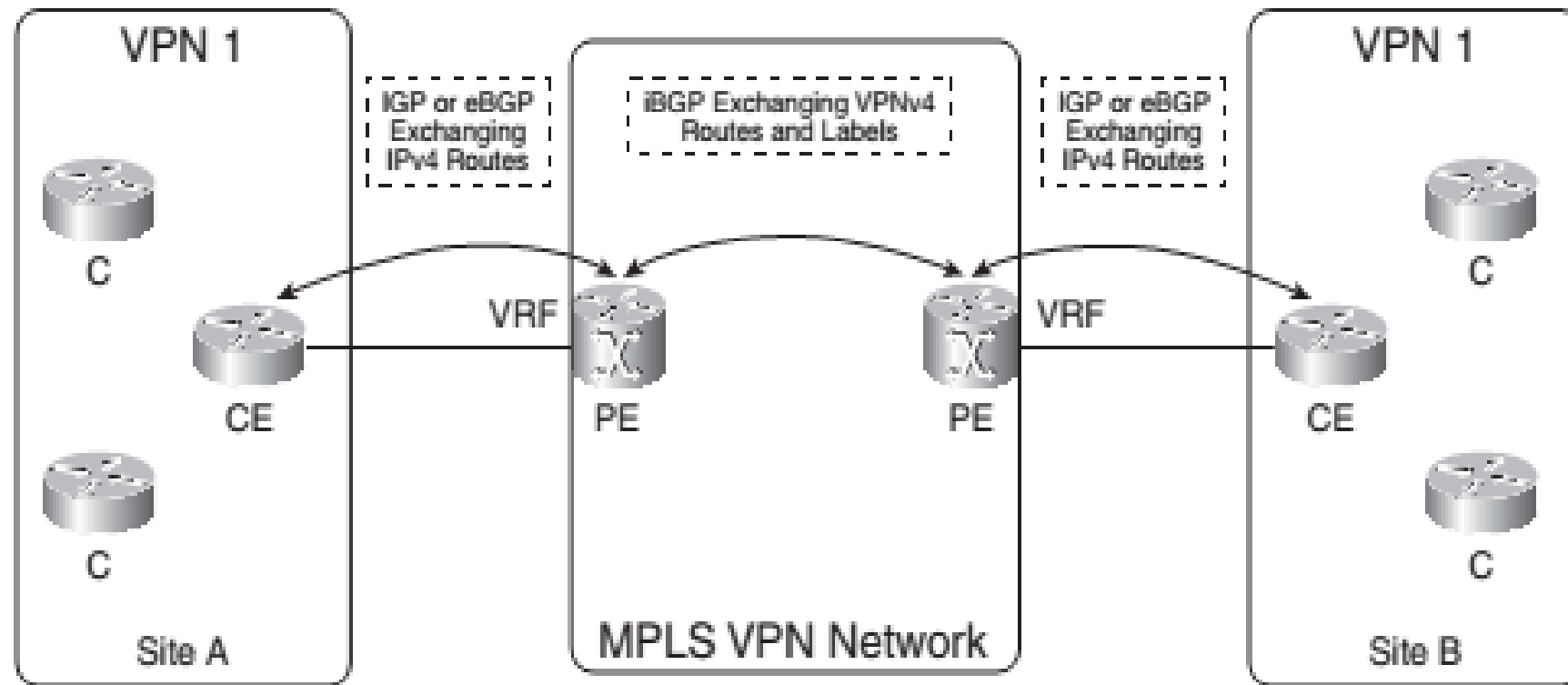
- En los routers PE, a estas rutas vpnv4, se le extrae el RD y son insertadas dentro de las routing tables de la VRF.
- Si la vpnv4 route es insertada en el VRF depende si el RT permite la importación de rutas en el VRF.
- Debido a que el service provider de la red MPLS VPN ejecuta BGP en un único sistema autónomo, entonces una instancia iBGP se ejecuta entre los routers PE.

## Propagación de rutas:

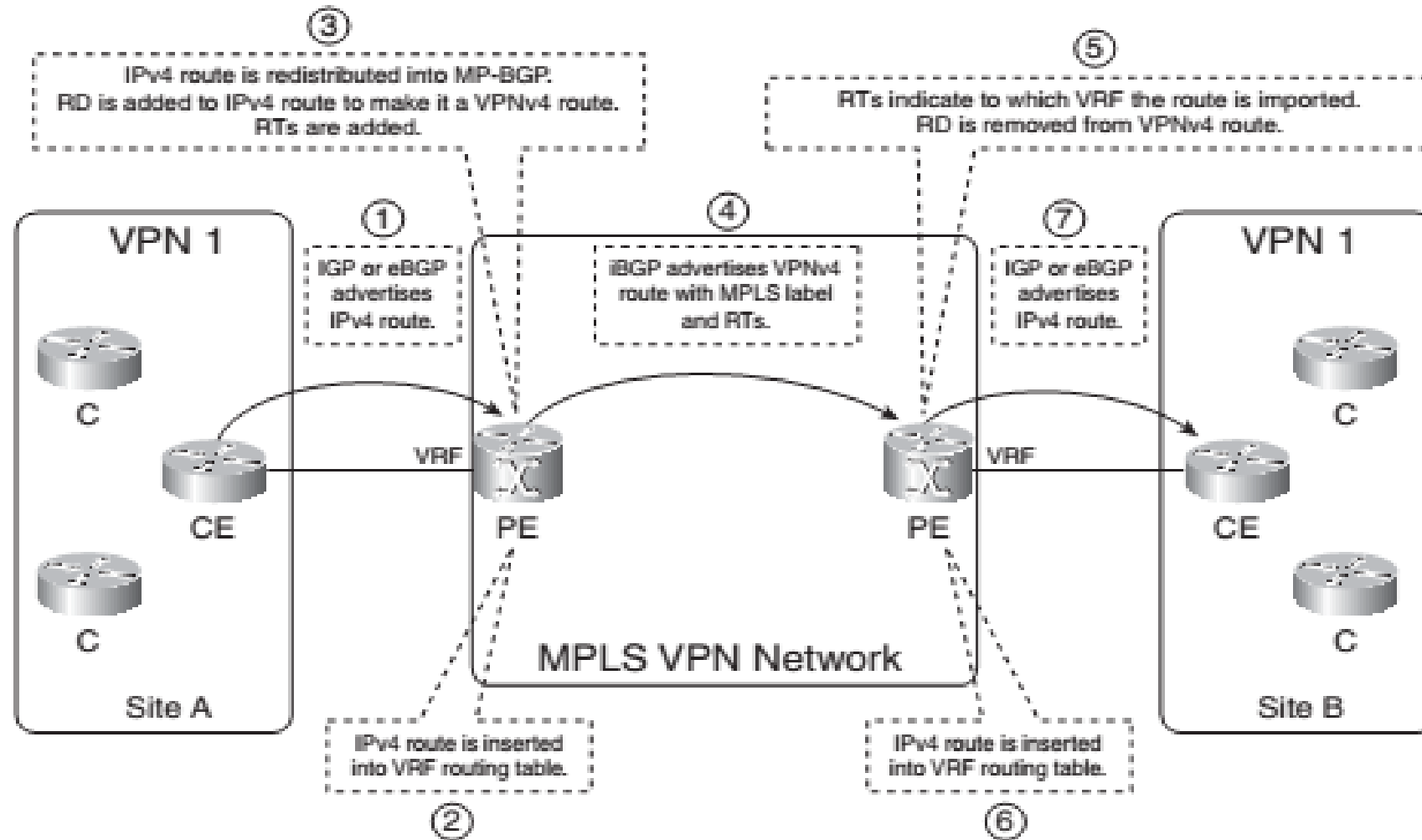
---

- La propagación desde eBGP (entre los CE y PE) a MP-iBGP es automática y no necesita configuración extra.
- Sin embargo, la redistribución IGP (entre los CE y PE) hacia MP-iBGP no es automática, por lo que se debe configurar manualmente.

# Propagación de rutas:

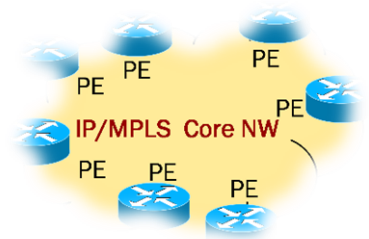


# Propagación de rutas:





# Packet Forwarding



## Packet Forwarding :

---

- Todos los paquetes de los clientes son forwardados con dos etiquetas:

- el IGP label

- el VPN label

# Packet Forwarding :

---

- El top label es el IGP label.
  - Este es distribuido a través de protocolos como LDP o RSVP-TE
  - Entre todos los routers P y PE salto por salto.
- El bottom label es el VPN label.
  - Este es publicado por MP-iBGP
  - Desde PE hacia el otro PE.

## Packet Forwarding :

---

- Los **routers P** utilizan el **label IGP**:
  - para forwardear el paquete al router PE de egreso correcto.
- El **router PE de egreso** utiliza el **VPN label**:
  - para forwardear el paquete al destino correcto.
- El VPN label de ser insertado por el router PE de ingreso para indicarle al router PE de egreso a cual VRF el paquete pertenece.

## Packet Forwarding :

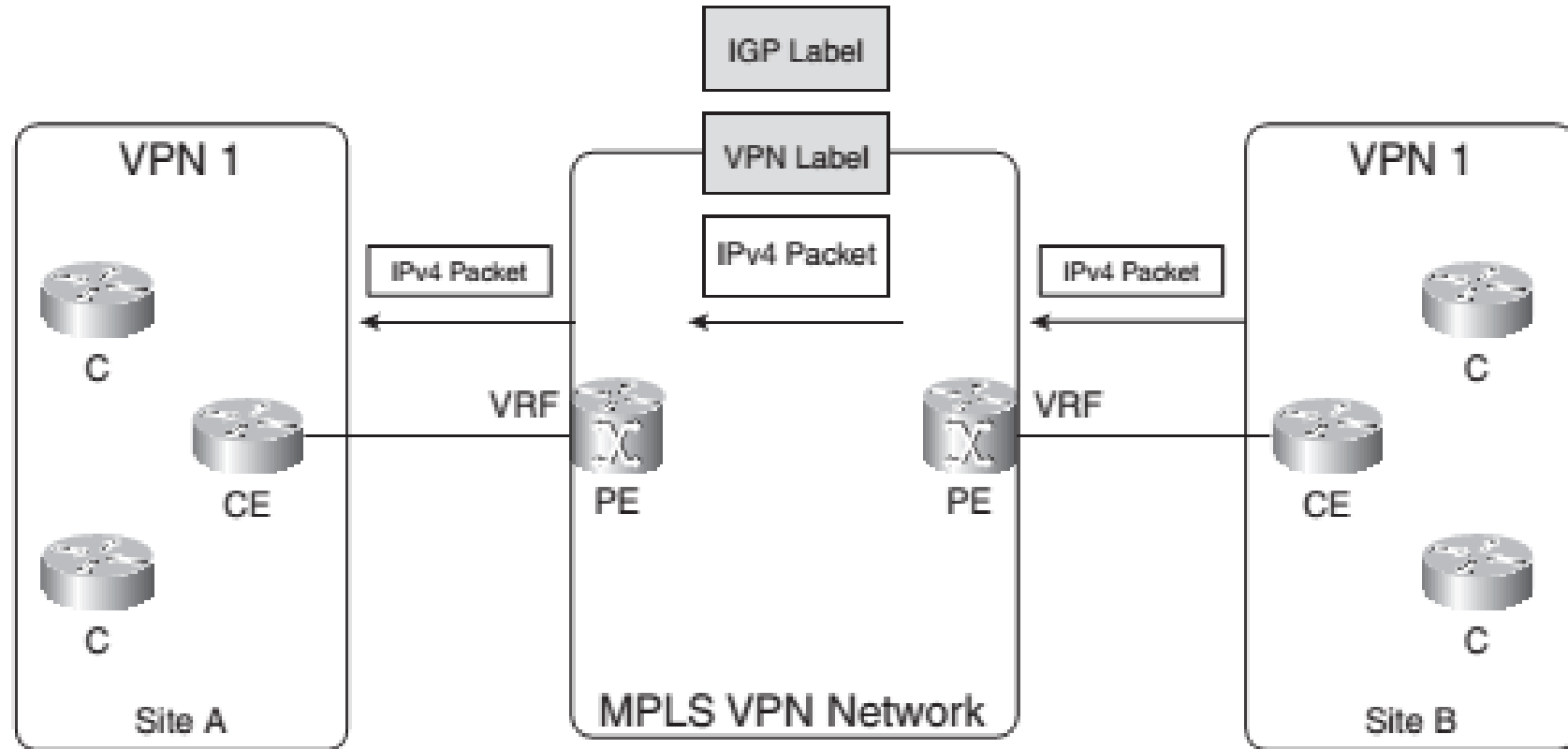
---

- Ahora:

Como hace el **router PE de egreso para señalar al router PE de ingreso** que label utilizar para un prefijo VRF determinado?

- Debido a que **MP-BGP** ya se utiliza para publicar los vpnv4 prefix, este **también señala el VPN label** que es asociado con el vpnv4 prefix

# Packet Forwarding :



# Packet Forwarding :

---

- En el siguiente ejemplo:
  - Intentaremos explicar el proceso que sucede en un paquete IP a medida que atraviesa el backbone MPLS VPN desde el sitio de un cliente hacia otro.
- Como primera instancia, **necesitamos los elementos necesarios** para poder construir la red MPLS VPN:
  - Multiprotocol iBGP,
    - tiene que ser ejecutado entre los routers PE que están distribuyendo las rutas vpnv4 y su VPN label asociado.

## Packet Forwarding :

---

- Un protocolo de distribución de etiquetas,
  - tiene que existir entre los routers PE y los routers P, como puede ser LDP o RSVP-TE.
- Entre los routers CE y los PE,
  - un protocolo de ruteo tiene que ser ejecutado para insertar las rutas del cliente en la routing table de la VRF en el router PE.
  - Finalmente, estas rutas deben ser distribuidas en MP-iBGP y viceversa.



## Packet Forwarding :

---

- Cuando **el paquete ingresa al router PE** de ingreso desde el CE,
- Este router PE realiza un **lookup de la destination IP address en el VRF** cust-one CEF table.
- El router PE de ingreso **determina el VRF correcto**,
  - en función de la configuración de la interfaz por la que entro el paquete en este router PE
  - determinando que VRF tiene asociada esta interfaz.

# Packet Forwarding :

---

- Primero, el router PE de ingreso realiza:
  - un push de la VPN label 30,
  - tal como fue publicada por BGP para el vpnv4 route.
  - Esta se convierte entonces en el bottom label.

## Packet Forwarding :

---

- Luego, el router mismo router PE realiza:
  - un push de la IGP label, como la top label.
  - En este caso realiza un push de la etiqueta con valor 16.
- Esta etiqueta es la etiqueta que está asociada con la ruta IGP /32 para el BGP next-hop IP address.
- Esta es usualmente la dirección IP de la interfaz de loopback en el router PE de egreso.
- En cada salto cambia o se realiza un swap del valor de la etiqueta.

## Packet Forwarding :

---

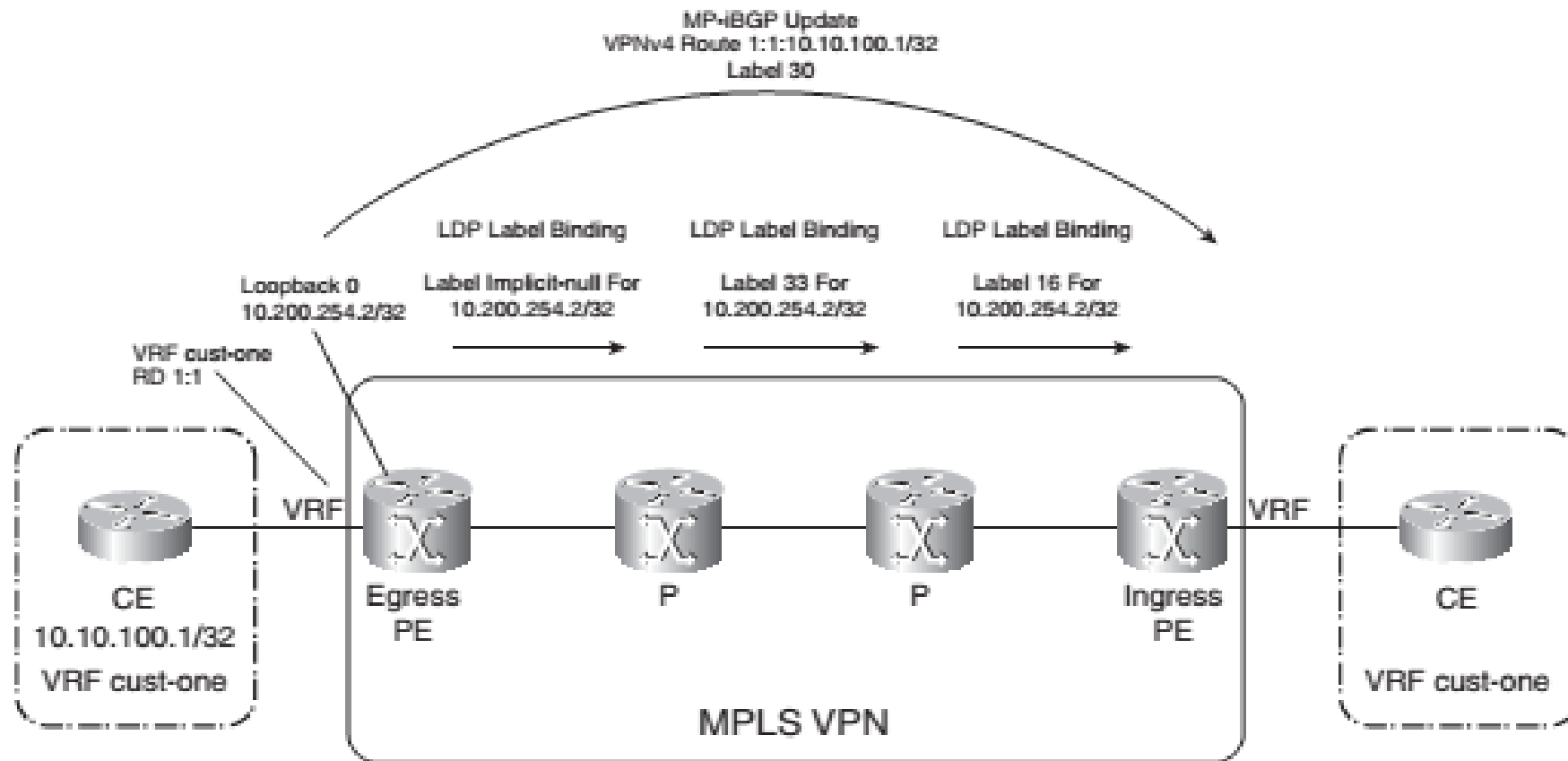
- El paquete IPv4 entonces **deja el router de ingreso PE con un stack de dos etiquetas.**
- La **etiqueta top label (la IGP label)** es intercambiada en cada salto a lo largo del camino.
- Esta etiqueta lleva al paquete hacia el router PE de egreso correcto.
- Por defecto, PHP está configurado** entre el último router P y el router PE de egreso.

## Packet Forwarding :

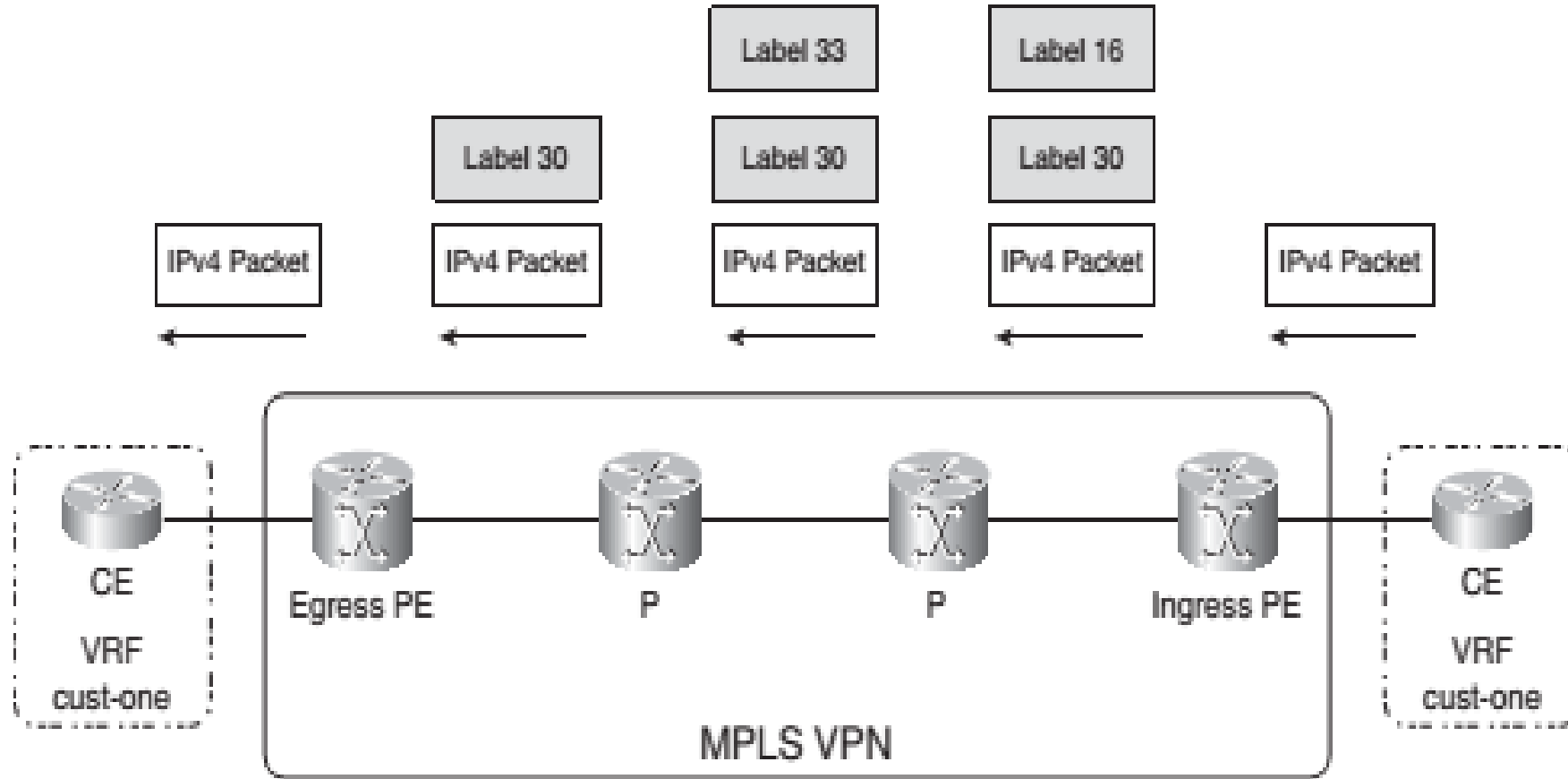
---

- Luego de esto, el IGP label es extraído o “popped off” en el último router P
- El paquete **ingresa al router PE de egreso únicamente con el VPN label**, del stack de dos etiquetas.
- Este router PE de egreso **revisa esta etiqueta VPN en la LFIB** y toma la decisión de forwarding.
- Luego, este **último label del stack es removido y el paquete es enviado al router CE** como un paquete IP puro.

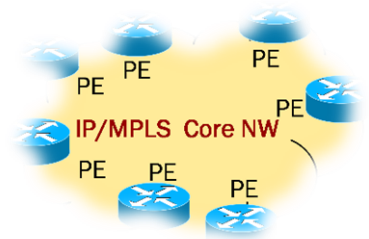
# Packet Forwarding :



# Packet Forwarding :

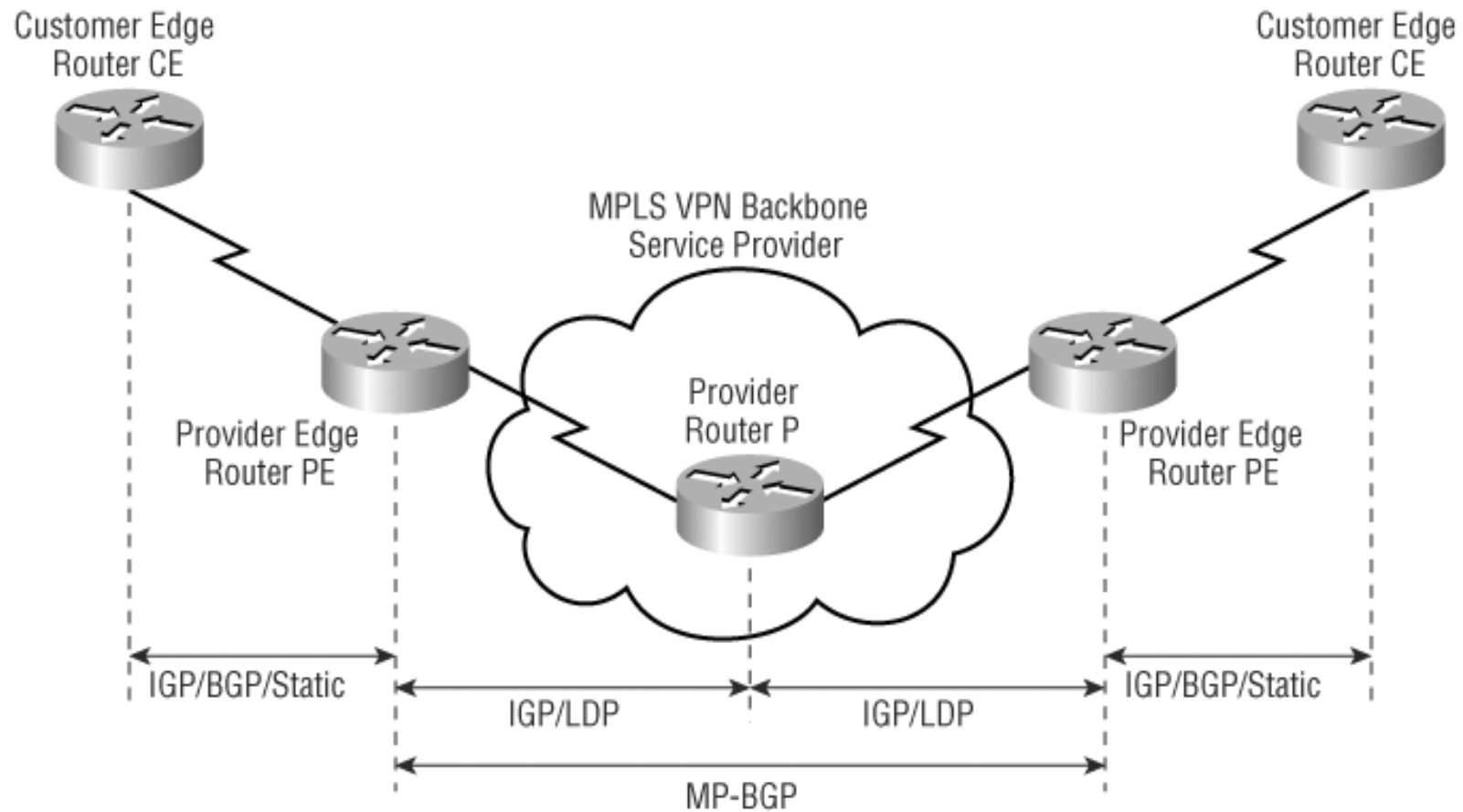


# Control Plane

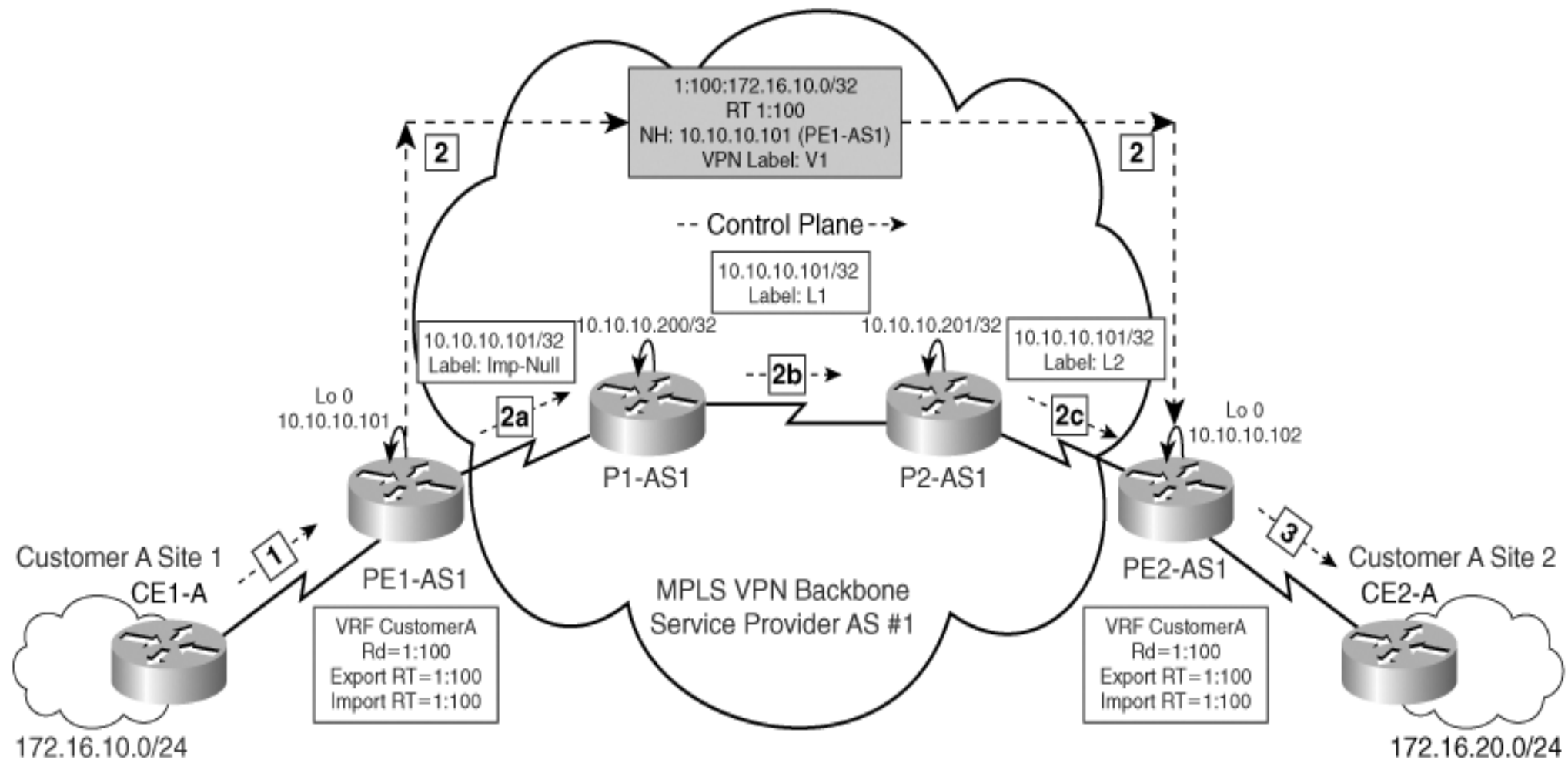




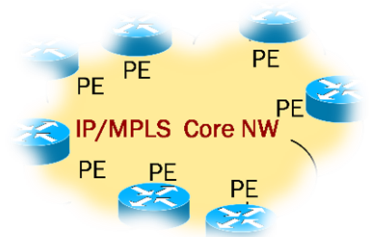
# Control Plane:



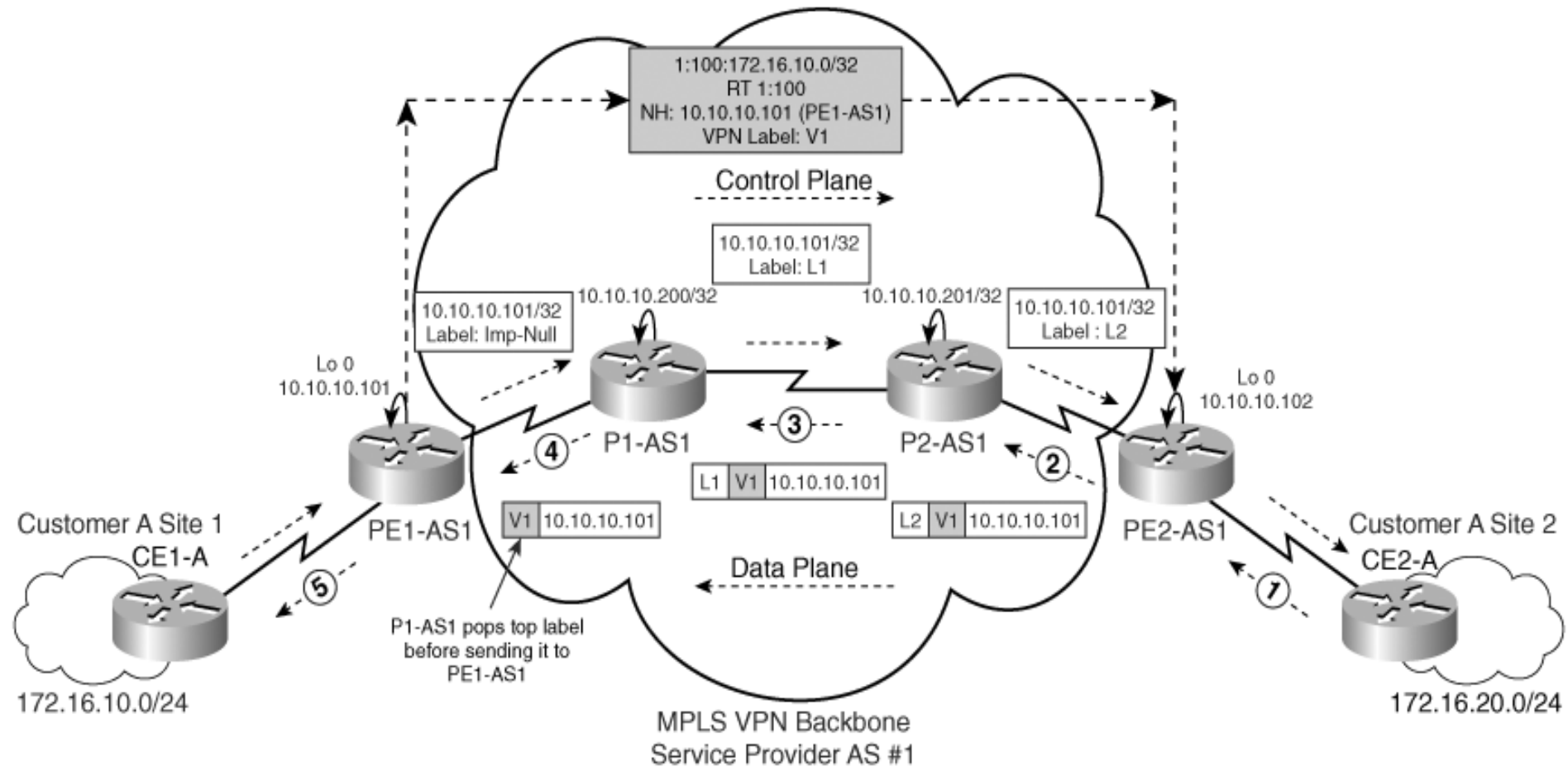
# Control Plane:



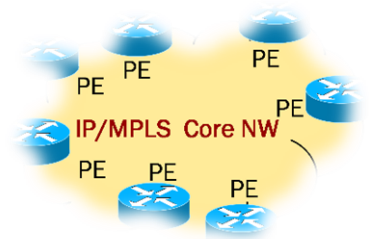
# Data Plane



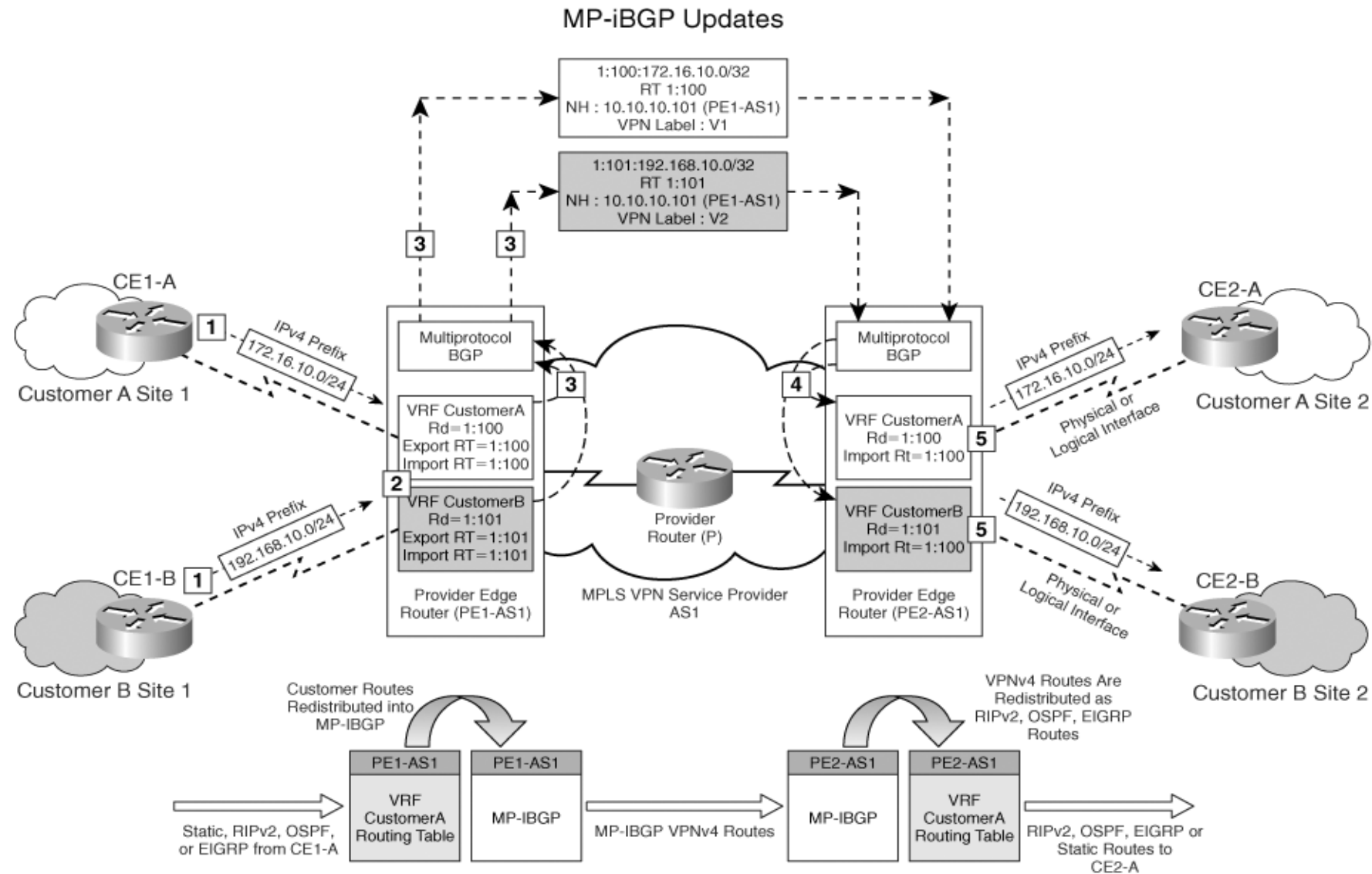
# Data Plane:



# Interacción RD / RT / MP-BGP



# Interacción RD / RT / MP-BGP:



## Interacción RD / RT / MP-BGP:

---

- El prefijo 172.16.10.0/24 es recibido en PE1 desde el router CE1-A.
- Este es parte del VRF CustomerA en el PE1-AS1.
- Luego PE1 realiza dos acciones:
  - Asocia un valor de RD de 1:100
  - Exporta un valor de RT de 1:100
- Todo esto tal lo definido en la configuración del VRF en el router PE1-AS1.

## Interacción RD / RT / MP-BGP:

---

- Luego de esto, las rutas aprendidas desde el router CE1-A son:
  - Redistribuidas dentro del MP-BGP en el PE1-AS1
  - donde el prefijo 172.16.10.0/24 es antepuesto con el valor RD 1:100
  - adjuntado con un route targeted extended community value (export RT) de 1:100
  - antes de enviar el prefijo VPNv4 como parte del MP-iBGP entre los routers PE.



## Interacción RD / RT / MP-BGP:

---

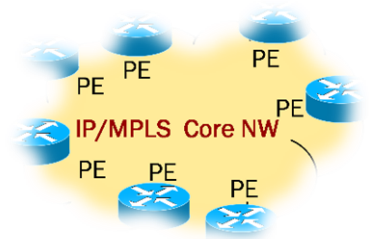
- El VPN label (3 bytes) es asignado a cada prefijo aprendido desde el proceso IGP del VRF del router CE conectado al router PE.
- El MP-BGP ejecutado dentro del dominio MPLS en la red del service provider transporta el VPNv4 prefix (IPv4 prefix + predended RD) en adición al BGP Route Target Extended Community.
- Adicionalmente, este valor de RT puede ser utilizado para realizar una importación selectiva de rutas dentro de un VRF cuando rutas VPNv4 son aprendidas mediante updates MP-iBGP.

## Interacción RD / RT / MP-BGP :

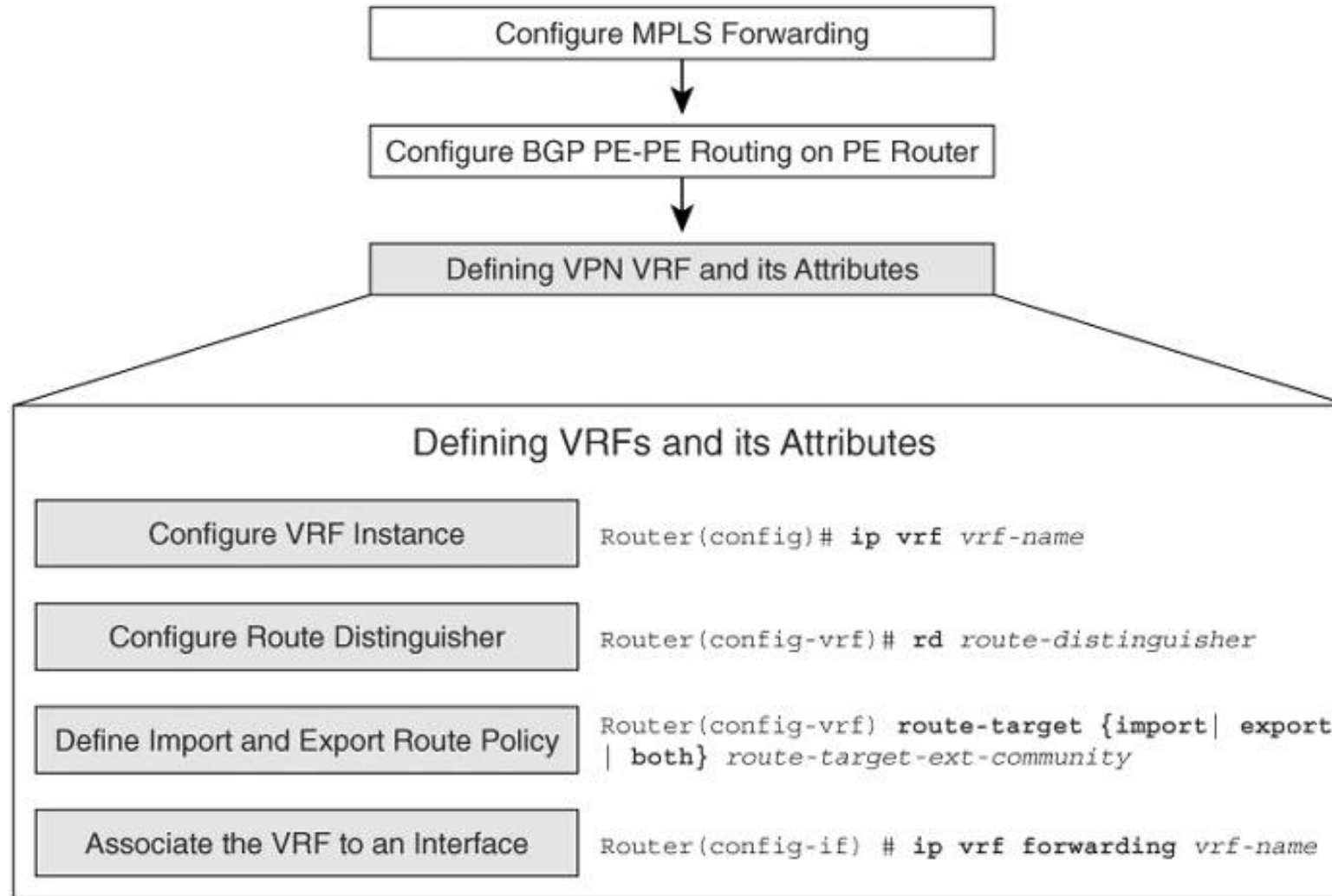
---

- El MP-BGP update son recibidos por el router PE2.
- La ruta es almacenada en la tabla VRF correspondiente para el Cliente A basada en el VPN label.
- Las rutas MP-BGP recibidas son redistribuidas dentro del proceso de routing VRF PE-CE y la ruta es propagada al CE2-A.

# Configuración



# Configuración:

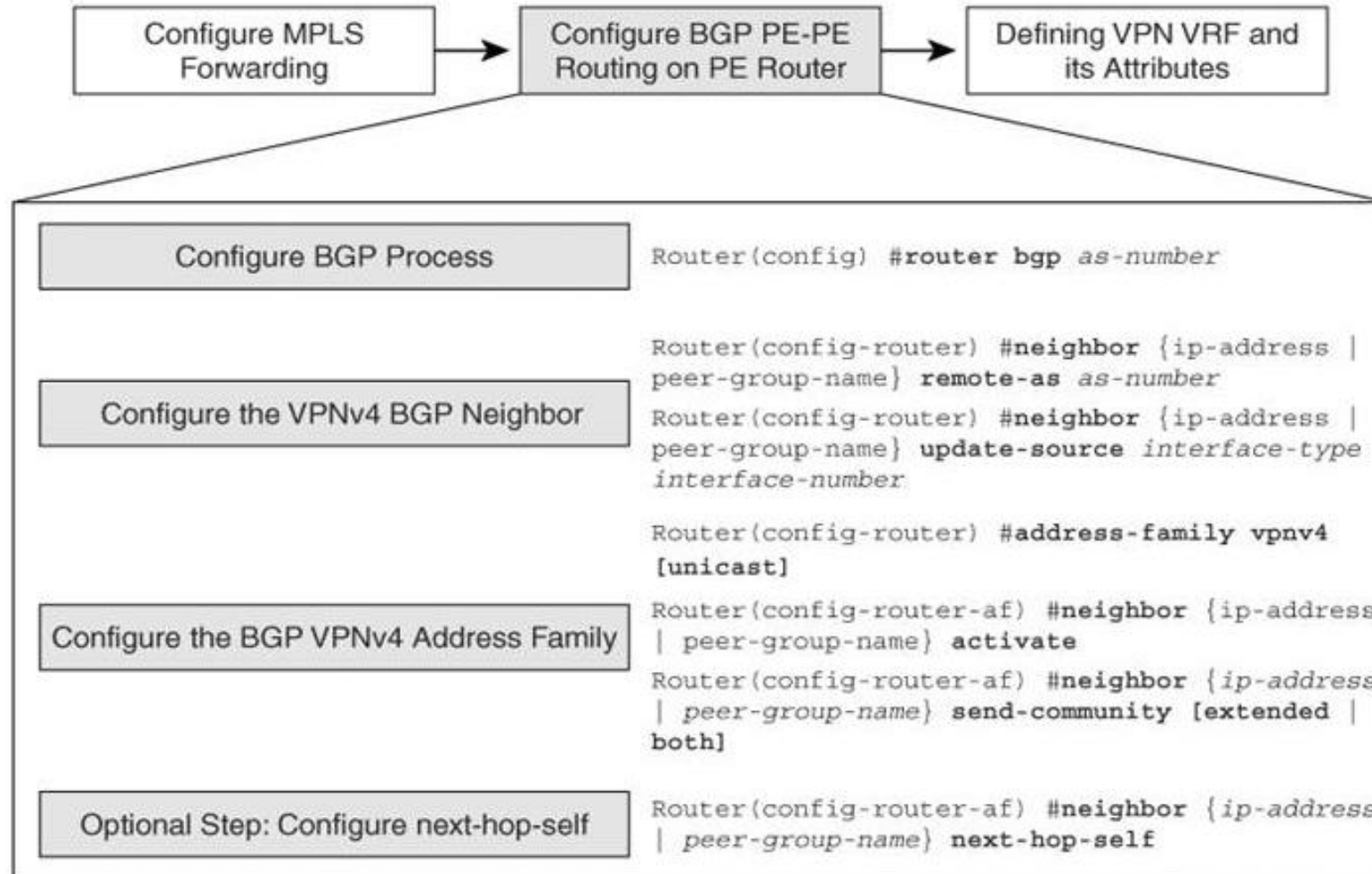


# Configuración:

---

```
ip vrf CustomerA
  rd 1:100
  route-target export 1:100
  route-target import 1:100
!
interface Serial1/0
  description PE-CE link to CE1-A
  ip vrf forwarding CustomerA
  ip address 172.16.1.1 255.255.255.0
!
interface Loopback1
  ip vrf forwarding CustomerA
  ip address 172.16.100.1 255.255.255.255
!
```

# Configuración:



# Configuración:

---

```
router bgp 1
  no synchronization
  neighbor 10.10.10.102 remote-as 1
  no auto-summary
  !
  address-family vpnv4
  neighbor 10.10.10.102 activate
  neighbor 10.10.10.102 send-community extended
  exit-address-family
  !
  address-family ipv4 vrf CustomerA
  redistribute connected
  no auto-summary
  no synchronization
  exit-address-family
```

# Configuración:

Command	Description
Router(config)#router bgp as-number	Configures the BGP routing process.
Router(config-router)#neighbor {ip-address   peer-group-name} remote-as as-number	Specifies a remote BGP neighbor to establish a BGP session.
Router(config-router)#neighbor {ip-address   ipv6-address   peer-group-name} update-source interface-type interface-number	Allows the BGP sessions to use any operational interface for TCP connections. The loopback interface is used frequently.
Router(config-router)#address-family vpnv4 [unicast]	Places the router in address family configuration mode, from which you can configure routing sessions that use VPN Version 4 address prefixes.
Router(config-router-af)#neighbor {ip-address   peer-group-name   ipv6-address} activate	Enables the exchange of information with a BGP neighboring router.
Router(config)#neighbor {ip-address   peer-group-name} next-hop-self	Configures the router as the next hop for a BGP-speaking neighbor or peer group.
Router#show ip bgp neighbors [neighbor-address] [received-routes   routes   advertised-routes   {paths regexp}   dampened-routes   received prefix-filter]	Displays information about the TCP and BGP connections to neighbors.
Router#show ip bgp summary	Displays the status of all BGP connections.
Router(config)#ip vrf vrf-name	Configures a VPN routing/forwarding instance (VRF) routing table.
Router(config-vrf)#rd route-distinguisher	Creates routing and forwarding tables for a VPN VRF.
Router(config-vrf)#route-target {import   export   both} route-target-ext-community	Creates a route target extended community for a VPN VRF. route-target-ext-community adds the route target extended community attributes to the VRF's list of import, export, or both (import and export) route target extended communities.
Router(config-if)#ip vrf forwarding vrf-name	Associates a VRF with an interface or subinterface.
Router#show ip vrf [brief   detail   interfaces   id] [vrf-name] [output-modifiers]	Displays the set of defined VPN VRFs and associated interfaces.
Router(config-router-af)#neighbor ip-address capability orf prefix-list [receive   send   both] or Router(config-router)# neighbor ip-address capability orf prefix-list [receive   send   both]	Enables the ORF capability to be sent as part of BGP open message and route-refresh messages to configured neighbors.
Router(config-router-af)# neighbor ip-address prefix-list prefix-list-name [in   out] or Router(config-router)# neighbor ip-address prefix-list prefix-list-name [in   out]	Associates a prefix list to a configured BGP neighbor.
Router(config)# ip prefix-list list-name [seq seq-value] {deny network/length   permit network/length}[ge ge-value] [le le-value]	Creates a prefix list with configured entries in which len < ge-value < le-value <= 32.
Router#show ip bgp vpnv4 {all   rd route-distinguisher   vrf vrf-name} [rib-failure] [ip-prefix/length [longer-prefixes] [output-modifiers]] [network-address [mask] [longer-prefixes] [output-modifiers]] [cidr-only] [community] [community-list] [dampened-paths] [filter-list] [flap-statistics] [inconsistent-as] [neighbors] [paths [line]] [peer-group] [quote-regexp] [regexp] [summary] [labels]	Displays VPN address information from the BGP table.



¿dudas?