Cisco IOS NAT - Integración con el MPLS VPN

Contenido

Introducción

El software del Network Address Translation (NAT) del ® del Cisco IOS permite el acceso a los servicios compartidos del MPLS VPNs múltiple, incluso cuando los dispositivos en los VPN utilizan los IP Addresses que solapan. Cisco IOS NAT está preparado para VRF y se puede configurar en los routers de borde del proveedor dentro de la red MPLS.

Nota: El MPLS en el IOS se soporta solamente con la herencia NAT. Ahora, no hay soporte en el Cisco IOS para NAT NVI con el MPLS.

El despliegue del MPLS VPNs se proyecta aumentar rápidamente durante los muchos años próximos. Las ventajas de una infraestructura de red común que la extensión rápida de los permisos y las opciones de conectividad flexibles indudablemente conducirán el crecimiento adicional en los servicios que pueden ser ofrecidos a la comunidad de la red interna.

Sin embargo, todavía sigue habiendo las barreras al crecimiento. El IPv6 y su promesa de un espacio de IP Address que exceda las necesidades de la Conectividad del futuro próximo sigue siendo en las fases tempranas de despliegue. De las redes existentes esquemas de direccionamiento del IP privado del uso comúnmente según lo definido dentro del RFC 1918 . La traducción de dirección de red es de uso frecuente interconectar las redes cuando los espacios de la dirección solapan o existe la duplicación.

Los proveedores de servicio y las empresas que tienen los servicios de la aplicación de red que quieren ofrecer o la parte con los clientes y los Partners querrán minimizar cualquier carga de la Conectividad puesta en el usuario del servicio. Es deseable, incluso obligatorio, ampliar el ofrecimiento a tantos usuarios posibles según las necesidades alcanzar las metas deseadas o volver. La necesidad funcionando del esquema de IP Addressing no ser una barrera que excluye a los usuarios posibles.

Desplegando el Cisco IOS NAT dentro de la infraestructura común del MPLS VPN, los proveedores de servicio de las comunicaciones pueden aliviar alguno de la carga de la Conectividad en los clientes y acelerar su capacidad de conectar servicios de aplicación compartidos a más consumidores de esos servicios.

Ventajas del NAT – Integración MPLS

La integración NAT con el MPLS tiene ventajas para ambos proveedores de servicio y sus clientes de Enterprise. Ofrece a proveedores de servicio más opciones para desplegar los servicios compartidos y para proporcionar el acceso a esos servicios. Las ofrendas del servicio adicional pueden ser un diferenciador sobre los competidores.

Para el proveedor de servicio	Para el VPN
Más ofertas de servicio	Costes reducidos
Opciones crecientes del acceso	Un acceso más simple
Ingresos crecientes	Dirección de la flexibilidad

Los clientes de Enterprise que intentan externalizar algo de su carga de trabajo actual pueden también beneficiarse de ofrendas más amplias por los proveedores de servicio. El desplazamiento de la carga de realizar cualquier traducción de la dirección necesaria a la red del proveedor de servicios los alivia de una tarea administrativa complicada. Los clientes pueden continuar utilizando el direccionamiento privado, con todo mantienen el acceso a los servicios compartidos y a Internet. La consolidación de la función NAT dentro de la red del proveedor de servicios puede también bajar el costo total a los clientes de Enterprise puesto que el Routers de la frontera del cliente no tiene que realizar la función NAT.

Aspectos del diseño

Al considerar los diseños que invocarán el NAT dentro de la red MPLS, el primer paso es determinar las necesidades del servicio desde un punto de vista de la aplicación. Usted necesitará considerar los protocolos comunicación usada y cualquier especial del cliente/del servidor impuesta por la aplicación. Aseegurese que el soporte necesario para los protocolos empleados es soportado y manejado por el Cisco IOS NAT. Una lista de protocolos admitidos se proporciona en los gatewayes de capa de aplicación del Cisco IOS NAT del documento.

Después, será necesario determinar el uso previsto del servicio compartido y las relaciones del tráfico anticipadas en las paquete-por-segundas. El NAT es una función Uso intensiva de la CPU del router. Por lo tanto, los requisitos de rendimiento serán un factor en la selección de una Opción de instrumentación determinada y determinarán el número de dispositivos NAT implicados.

También, considere cualesquiera problemas de seguridad y precaución que deban ser tomados. Aunque el MPLS VPNs, por definición, sea privado y con eficacia el tráfico separado, la red de servicio compartida es generalmente común entre muchos VPN.

Escenarios de instrumentación

Hay dos opciones para el despliegue NAT dentro del borde del proveedor MPLS:

- Centralizado con la salida NAT PE
- Distribuido con el ingreso NAT PE

Algunas ventajas a configurar la función NAT en el punto de egreso de la red MPLS lo más cerca posible a la red de servicio compartida incluyen:

- Una configuración centralizada que promueve un aprovisionamiento más simple del servicio
- Troubleshooting simplificado
- Scalability operativo aumentado
- Requisitos disminuidos de la asignación de IP Address

Sin embargo, las ventajas son compensadas por una reducción en el scalability y el funcionamiento. Éste es el equilibrio principal que debe ser considerado. Por supuesto, la función NAT se puede también realizar dentro de las redes del cliente si se determina que la integración de esta característica con una red MPLS no es deseable.

Ingreso PE NAT

El NAT se puede configurar en el router del ingreso PE de la red MPLS tal y como se muestra en del cuadro 1. Con este diseño, el scalability se mantiene en gran parte mientras que el funcionamiento es optimizado distribuyendo la función NAT sobre muchos dispositivos de borde. Cada NAT PE maneja el tráfico para los sitios localmente conectados con ese PE. Reglas NAT y listas de control de acceso o control de los mapa del ruta que los paquetes requieren la traducción.

Figura 1: Ingreso PE NAT

Hay una restricción que previene el NAT entre dos VRF mientras que también proporciona al NAT a un servicio compartido tal y como se muestra en del cuadro 2. Esto es debido al requisito de señalar las interfaces como del “exterior” NAT interfaces del “interior” y. El soporte para las conexiones entre los VRF en un solo PE se planea para un Cisco IOS Release futuro.

Figura 2: Interempresarial

Salida PE NAT

El NAT se puede configurar en el router de la salida PE de la red MPLS tal y como se muestra en del cuadro 3. Con este diseño, el scalability se reduce a un cierto grado puesto que el PE central debe mantener las rutas para todas las redes del cliente que accedan el servicio compartido. Los requisitos de rendimiento de la aplicación deben también ser considerados de modo que el tráfico no sobrecargue al router que debe traducir los IP Addresses de los paquetes. Porque el NAT ocurre centralmente para todos los clientes que usan esta trayectoria, los pools de la dirección IP pueden ser compartidos; así, el número total de subredes requeridas se reduce.

Figura 3: Salida PE NAT

Los routeres múltiples podrían ser desplegados para aumentar el scalability del diseño de la salida PE NAT tal y como se muestra en del cuadro 4. En este escenario, el cliente VPN podría ser “aprovisionado” en un router NAT específico. La traducción de dirección de red ocurriría para el tráfico total a y desde el servicio compartido para eso fijó de los VPN. Por ejemplo, el tráfico de los VPN para el cliente A y B podría utilizar el NAT-PE1, mientras que el tráfico a y desde el VPN para el C del cliente utiliza el NAT-PE2. Cada NAT PE llevaría el tráfico solamente para los VPN específicos definidos y mantendría solamente las rutas de nuevo a los sitios en esos VPN. Los

conjuntos de direcciones NAT separados podrían ser definidos dentro de cada uno del Routers NAT PE para rutear los paquetes de la red de servicio compartida al NAT apropiado PE para la traducción y la encaminamiento de nuevo al cliente VPN.

Figura 4: Salida múltiple PE NAT

El diseño centralizado impone una restricción ante cómo la red de servicio compartida debe ser configurada. Específicamente, el uso de la importación/de la exportación de las rutas del MPLS VPN entre un servicio compartido VPN y el cliente VPN no es posibles. Esto es debido a la naturaleza de la operación MPLS según lo especificado por el RFC 2547 . Cuando las rutas se importan y se exportan usando las comunidades ampliadas y los descriptores de Route, el NAT no puede determinar la fuente VPN del paquete que entra en el NAT central PE. El caso usual es hacer la red de servicio compartida una interfaz genérica bastante que una interfaz VRF. Una ruta a la red de servicio compartida entonces se agrega en el NAT central PE para cada tabla VRF asociada a un cliente VPN que necesita el acceso al servicio compartido como parte del proceso de abastecimiento. Esto se describe más detalladamente más adelante.

Opciones de instrumentación y detalles de la configuración

Esta sección incluye algunos detalles relacionados con cada uno de las Opciones de instrumentación. Los ejemplos todos se toman de la red mostrada en el cuadro 5. refieren a este diagrama para el resto de esta sección.

Nota: En la red usada para ilustrar la operación de VRF NAT para este papel, solamente el Routers PE es incluido. No hay Routers de la base “P”. Sin embargo, los mecanismos esenciales pueden todavía ser considerados.

Figura 5: Ejemplo de la configuración del NAT VRF

Salida PE NAT

En este ejemplo, el **Gila** y el **dragón** marcados los routeres de borde del proveedor se configuran como Routers simple PE. El PE central cerca del servicio compartido LAN (**iguana**) se configura para el NAT. A cada cliente VPN comparte a un solo agrupamiento NAT que necesita el acceso al servicio compartido. El NAT se realiza solamente en los paquetes destinados para el host compartido del servicio en 88.1.88.8.

Reenvío de datos de la salida PE NAT

Con el MPLS, cada paquete ingresa la red en un ingreso PE y sale la red MPLS en una salida PE. La trayectoria del Routers del switching por etiquetas atravesado del ingreso a la salida se conoce como la trayectoria conmutada de etiquetas (LSP). El LSP es unidireccional. Un diverso LSP se utiliza para el tráfico de retorno.

Al usar la salida PE NAT, un Forwarding Equivalence Class (FEC) se define con eficacia para todo el tráfico de los usuarios del servicio compartido. Es decir todos los paquetes destinados para el servicio compartido LAN son miembros de un FEC común. Un paquete se asigna a un FEC determinado apenas una vez en el borde de acceso de la red y sigue el LSP a la salida PE. El FEC es señalado en el paquete de datos agregando una escritura de la etiqueta determinada.

Flujo de paquetes al servicio compartido del VPN

Para que los dispositivos en los VPN múltiples que tienen esquemas de la dirección superpuesta

para acceder un host compartido del servicio, se requiere el NAT. Cuando el NAT se configura en la salida PE, las entradas de tabla de la traducción de dirección de red incluirán un identificador VRF para distinguir a las direcciones duplicadas y para asegurar la encaminamiento apropiada.

Figura 6: Paquetes transmitidos a la salida PE NAT

El cuadro 6 ilustra los paquetes destinados para un host compartido del servicio a partir el dos del cliente VPN que tienen esquemas de direccionamiento del IP duplicado. La figura muestra que un paquete que originaba en el cliente A con una dirección de origen de 172.31.1.1 destinó para un servidor compartido en 88.1.88.8. Otro paquete del cliente B con la misma dirección IP de origen también se envía al mismo servidor compartido. Cuando los paquetes alcanzan al router PE, las operaciones de búsqueda de la capa 3 se hacen para la red del IP de destino en la Base de información de reenvío (FIB).

La entrada de la BOLA dice al router PE remitir el tráfico a la salida PE usando una pila de etiquetas. La etiqueta inferior en el stack es asignada por el router del destino PE, en este caso **iguana del** router.

```
iguana# show ip cef vrf custA 88.1.88.8 88.1.88.8/32, version 47, epoch 0, cached adjacency
88.1.3.2 0 packets, 0 bytes tag information set local tag: VPN-route-head fast tag rewrite with
Et1/0, 88.1.3.2, tags imposed: {24} via 88.1.11.5, 0 dependencies, recursive next hop 88.1.3.2,
Ethernet1/0 via 88.1.11.5/32 valid cached adjacency tag rewrite with Et1/0, 88.1.3.2, tags
imposed: {24} iguana# show ip cef vrf custB 88.1.88.8 88.1.88.8/32, version 77, epoch 0, cached
adjacency 88.1.3.2 0 packets, 0 bytes tag information set local tag: VPN-route-head fast tag
rewrite with Et1/0, 88.1.3.2, tags imposed: {28} via 88.1.11.5, 0 dependencies, recursive next
hop 88.1.3.2, Ethernet1/0 via 88.1.11.5/32 valid cached adjacency tag rewrite with Et1/0,
88.1.3.2, tags imposed: {28} iguana#
```

Podemos ver de la visualización que los paquetes del custA VRF tendremos un valor de la etiqueta de 24 (0x18) y los paquetes del custB VRF tendrán un valor de la etiqueta de 28 (0x1C).

En este caso, porque no hay Routers “P” en nuestra red, allí no es etiqueta adicional impuesta. Había habido routeres del núcleo, una escritura de la etiqueta exterior habría sido impuesta y el proceso normal del intercambio de la escritura de la etiqueta habría ocurrido dentro de la red del núcleo hasta que el paquete alcanzara la salida PE.

Puesto que el router del **Gila** está conectado directamente con la salida PE, vemos que la etiqueta está hecha estallar antes de que se agregue nunca:

```
gila# show tag-switching forwarding-table Local Outgoing Prefix Bytes tag Outgoing Next Hop tag
tag or VC or Tunnel Id switched interface 16 Pop tag 88.1.1.0/24 0 Et1/1 88.1.2.2 Pop tag
88.1.1.0/24 0 Et1/0 88.1.3.2 17 Pop tag 88.1.4.0/24 0 Et1/1 88.1.2.2 18 Pop tag 88.1.10.0/24 0
Et1/1 88.1.2.2 19 Pop tag 88.1.11.1/32 0 Et1/1 88.1.2.2 20 Pop tag 88.1.5.0/24 0 Et1/0 88.1.3.2
21 19 88.1.11.10/32 0 Et1/1 88.1.2.2 22 88.1.11.10/32 0 Et1/0 88.1.3.2 22 20 172.18.60.176/32 0
Et1/1 88.1.2.2 23 172.18.60.176/32 0 Et1/0 88.1.3.2 23 Untagged 172.31.1.0/24[V] 4980 Fa0/0
10.88.162.6 24 Aggregate 10.88.162.4/30[V] 1920 25 Aggregate 10.88.162.8/30[V] 137104 26
Untagged 172.31.1.0/24[V] 570 Et1/2 10.88.162.14 27 Aggregate 10.88.162.12/30[V] \ 273480 30 Pop
tag 88.1.11.5/32 0 Et1/0 88.1.3.2 31 Pop tag 88.1.88.0/24 0 Et1/0 88.1.3.2 32 16 88.1.97.0/24 0
Et1/0 88.1.3.2 33 Pop tag 88.1.99.0/24 0 Et1/0 88.1.3.2 gila# gila# show tag-switching
forwarding-table 88.1.88.0 detail Local Outgoing Prefix Bytes tag Outgoing Next Hop tag tag or
VC or Tunnel Id switched interface 31 Pop tag 88.1.88.0/24 0 Et1/0 88.1.3.2 MAC/Encaps=14/14,
MRU=1504, Tag Stack{} 005054D92A250090BF9C6C1C8847 No output feature configured Per-packet load-
sharing gila#
```

Las visualizaciones siguientes representan los paquetes de eco según lo recibido por el router NAT de la salida PE (en la interfaz E1/0/5 en la **iguana**).

```
From CustA: DLC: ----- DLC Header ----- DLC: DLC: Frame 1 arrived at 16:21:34.8415; frame size
is 118 (0076 hex) bytes. DLC: Destination = Station 005054D92A25 DLC: Source = Station
```

0090BF9C6C1C DLC: Ethertype = 8847 (MPLS) DLC: MPLS: ----- MPLS Label Stack ----- MPLS: **MPLS: Label Value = 00018** MPLS: Reserved For Experimental Use = 0 MPLS: Stack Value = 1 (Bottom of Stack) MPLS: Time to Live = 254 (hops) MPLS: IP: ----- IP Header ----- IP: IP: Version = 4, header length = 20 bytes IP: Type of service = 00 IP: 000. .... = routine IP: ...0 .... = normal delay IP: .... 0... = normal throughput IP: .... .0.. = normal reliability IP: .... ..0. = ECT bit - transport protocol will ignore the CE bit IP: ....0 = CE bit - no congestion IP: Total length = 100 bytes IP: Identification = 175 IP: Flags = 0X IP: .0.. .... = may fragment IP: ..0. .... = last fragment IP: Fragment offset = 0 bytes IP: Time to live = 254 seconds/hops IP: Protocol = 1 (ICMP) IP: Header checksum = 5EC0 (correct) **IP: Source address = [172.31.1.1]** IP: Destination address = [88.1.88.8] IP: No options IP: ICMP: ----- ICMP header ----- ICMP: ICMP: Type = 8 (Echo) ICMP: Code = 0 ICMP: Checksum = 4AF1 (correct) ICMP: Identifier = **4713** ICMP: Sequence number = 6957 ICMP: [72 bytes of data] ICMP: ICMP: [Normal end of "ICMP header".]
From CustB: DLC: ----- DLC Header ----- DLC: DLC: Frame 11 arrived at 16:21:37.1558; frame size is 118 (0076 hex) bytes. DLC: Destination = Station 005054D92A25 DLC: Source = Station 0090BF9C6C1C DLC: Ethertype = 8847 (MPLS) DLC: MPLS: ----- MPLS Label Stack ----- MPLS: **MPLS: Label Value = 0001C** MPLS: Reserved For Experimental Use = 0 MPLS: Stack Value = 1 (Bottom of Stack) MPLS: Time to Live = 254 (hops) MPLS: IP: ----- IP Header ----- IP: IP: Version = 4, header length = 20 bytes IP: Type of service = 00 IP: 000. .... = routine IP: ...0 .... = normal delay IP: .... 0... = normal throughput IP: .... .0.. = normal reliability IP: .... ..0. = ECT bit - transport protocol will ignore the CE bit IP: ....0 = CE bit - no congestion IP: Total length = 100 bytes IP: Identification = 165 IP: Flags = 0X IP: .0.. .... = may fragment IP: ..0. .... = last fragment IP: Fragment offset = 0 bytes IP: Time to live = 254 seconds/hops IP: Protocol = 1 (ICMP) IP: Header checksum = 5ECA (correct) **IP: Source address = [172.31.1.1]** IP: Destination address = [88.1.88.8] IP: No options IP: ICMP: ----- ICMP header ----- ICMP: ICMP: Type = 8 (Echo) ICMP: Code = 0 ICMP: Checksum = AD5E (correct) ICMP: Identifier = **3365** ICMP: Sequence number = 7935 ICMP: [72 bytes of data] ICMP: ICMP: [Normal end of "ICMP header".]

Estos ping dan lugar a las entradas siguientes que son creadas en la tabla NAT en la iguana del router de la salida PE. Las entradas específicas creadas para los paquetes mostrados arriba se pueden corresponder con por su identificador ICMP.

iguana# **show ip nat translations** Pro Inside global Inside local Outside local Outside global **icmp 192.168.1.3:3365 172.31.1.1:3365 88.1.88.8:3365 88.1.88.8:3365** icmp 192.168.1.3:3366 172.31.1.1:3366 88.1.88.8:3366 88.1.88.8:3366 icmp 192.168.1.3:3367 172.31.1.1:3367 88.1.88.8:3367 88.1.88.8:3367 icmp 192.168.1.3:3368 172.31.1.1:3368 88.1.88.8:3368 88.1.88.8:3368 icmp 192.168.1.3:3369 172.31.1.1:3369 88.1.88.8:3369 88.1.88.8:3369 **icmp 192.168.1.1:4713 172.31.1.1:4713 88.1.88.8:4713 88.1.88.8:4713** icmp 192.168.1.1:4714 172.31.1.1:4714 88.1.88.8:4714 88.1.88.8:4714 icmp 192.168.1.1:4715 172.31.1.1:4715 88.1.88.8:4715 88.1.88.8:4715 icmp 192.168.1.1:4716 172.31.1.1:4716 88.1.88.8:4716 88.1.88.8:4716 icmp 192.168.1.1:4717 172.31.1.1:4717 88.1.88.8:4717 88.1.88.8:4717 iguana# **show ip nat translations** *verbose* Pro Inside global Inside local Outside local Outside global icmp 192.168.1.3:3365 172.31.1.1:3365 88.1.88.8:3365 88.1.88.8:3365 create 00:00:34, use 00:00:34, left 00:00:25, Map-Id(In): 2, flags: extended, use_count: 0, VRF : **custB** icmp 192.168.1.3:3366 172.31.1.1:3366 88.1.88.8:3366 88.1.88.8:3366 create 00:00:34, use 00:00:34, left 00:00:25, Map-Id(In): 2, flags: extended, use_count: 0, VRF : **custB** icmp 192.168.1.3:3367 172.31.1.1:3367 88.1.88.8:3367 88.1.88.8:3367 create 00:00:34, use 00:00:34, left 00:00:25, Map-Id(In): 2, flags: extended, use_count: 0, VRF : **custB** icmp 192.168.1.3:3368 172.31.1.1:3368 88.1.88.8:3368 88.1.88.8:3368 create 00:00:34, use 00:00:34, left 00:00:25, Map-Id(In): 2, flags: extended, use_count: 0, VRF : **custB** icmp 192.168.1.3:3369 172.31.1.1:3369 88.1.88.8:3369 88.1.88.8:3369 create 00:00:34, use 00:00:34, left 00:00:25, Map-Id(In): 2, flags: extended, use_count: 0, VRF : **custB** icmp 192.168.1.1:4713 172.31.1.1:4713 88.1.88.8:4713 88.1.88.8:4713 create 00:00:37, use 00:00:37, left 00:00:22, Map-Id(In): 1, Pro Inside global Inside local Outside local Outside global flags: extended, use_count: 0, VRF : **custA** icmp 192.168.1.1:4714 172.31.1.1:4714 88.1.88.8:4714 88.1.88.8:4714 create 00:00:37, use 00:00:37, left 00:00:22, Map-Id(In): 1, flags: extended, use_count: 0, VRF : **custA** icmp 192.168.1.1:4715 172.31.1.1:4715 88.1.88.8:4715 88.1.88.8:4715 create 00:00:37, use 00:00:37, left 00:00:22, Map-Id(In): 1, flags: extended, use_count: 0, VRF : **custA** icmp 192.168.1.1:4716 172.31.1.1:4716 88.1.88.8:4716 88.1.88.8:4716 create 00:00:37, use 00:00:37, left 00:00:22, Map-Id(In): 1, flags: extended, use_count: 0, VRF : **custA** icmp 192.168.1.1:4717 172.31.1.1:4717 88.1.88.8:4717 88.1.88.8:4717 create 00:00:37, use 00:00:37, left 00:00:22, Map-Id(In): 1, flags: extended, use_count: 0, VRF : **custA** iguana#

Flujo de paquetes del servicio compartido de nuevo al origen VPN

Mientras que los paquetes fluyen de nuevo a los dispositivos que han accedido el host compartido

del servicio, la tabla NAT se examina antes de la encaminamiento (paquetes que van de la interfaz del “exterior” NAT a la interfaz del “interior”). Porque cada entrada única incluye el identificador correspondiente VRF, el paquete se puede traducir y rutear apropiadamente.

Figura 7: Paquetes transmitidos de nuevo al usuario del servicio compartido

Tal y como se muestra en del cuadro 7, el tráfico de retorno primero es examinado por el NAT para encontrar una entrada de traducción que corresponde con. Por ejemplo, un paquete se envía al destino 192.168.1.1. Se busca la tabla NAT. Cuando se encuentra la coincidencia, la traducción adecuada se hace al direccionamiento del “Inside Local” (172.31.1.1) y entonces las operaciones de búsqueda de la adyacencia se realizan usando el VRF asociado ID de la entrada de NAT.

```
iguana# show ip cef vrf custA 172.31.1.0 172.31.1.0/24, version 12, epoch 0, cached adjacency
88.1.3.1 0 packets, 0 bytes tag information set local tag: VPN-route-head fast tag rewrite with
Et1/0/5, 88.1.3.1, tags imposed: {23} via 88.1.11.9, 0 dependencies, recursive next hop
88.1.3.1, Ethernet1/0/5 via 88.1.11.9/32 valid cached adjacency tag rewrite with Et1/0/5,
88.1.3.1, tags imposed: {23} iguana# show ip cef vrf custB 172.31.1.0 172.31.1.0/24, version 18,
epoch 0, cached adjacency 88.1.3.1 0 packets, 0 bytes tag information set local tag: VPN-route-
head fast tag rewrite with Et1/0/5, 88.1.3.1, tags imposed: {26} via 88.1.11.9, 0 dependencies,
recursive next hop 88.1.3.1, Ethernet1/0/5 via 88.1.11.9/32 valid cached adjacency tag rewrite
with Et1/0/5, 88.1.3.1, tags imposed: {26} iguana#
```

La escritura de la etiqueta 23 (0x17) se utiliza para el tráfico destinado para 172.31.1.0/24 en el custA y la escritura de la etiqueta 26 (0x1A) VRF se utiliza para los paquetes destinados para 172.31.1.0/24 en el custB VRF.

Esto se ve en los paquetes de respuesta de eco enviados de la **iguana del** router:

```
To custA: DLC: ----- DLC Header ----- DLC: DLC: Frame 2 arrived at 16:21:34.8436; frame size is
118 (0076 hex) bytes. DLC: Destination = Station 0090BF9C6C1C DLC: Source = Station 005054D92A25
DLC: Ethertype = 8847 (MPLS) DLC: MPLS: ----- MPLS Label Stack ----- MPLS: MPLS: Label Value =
00017 MPLS: Reserved For Experimental Use = 0 MPLS: Stack Value = 1 (Bottom of Stack) MPLS: Time
to Live = 254 (hops) MPLS: IP: ----- IP Header ----- IP: IP: Version = 4, header length = 20
bytes IP: Type of service = 00 IP: 000. .... = routine IP: ...0 .... = normal delay IP: ....
0... = normal throughput IP: .... .0.. = normal reliability IP: .... ..0. = ECT bit - transport
protocol will ignore the CE bit IP: .... ...0 = CE bit - no congestion IP: Total length = 100
bytes IP: Identification = 56893 IP: Flags = 4X IP: .1.. .... = don't fragment IP: ..0. .... =
last fragment IP: Fragment offset = 0 bytes IP: Time to live = 254 seconds/hops IP: Protocol = 1
(ICMP) IP: Header checksum = 4131 (correct) IP: Source address = [88.1.88.8] IP: Destination
address = [172.31.1.1] IP: No options IP: ICMP: ----- ICMP header ----- ICMP: ICMP: Type = 0
(Echo reply) ICMP: Code = 0 ICMP: Checksum = 52F1 (correct) ICMP: Identifier = 4713 ICMP:
Sequence number = 6957 ICMP: [72 bytes of data] ICMP: ICMP: [Normal end of "ICMP header".]
```

Cuando el paquete alcanza al router del destino PE, la escritura de la etiqueta se utiliza para determinar el VRF apropiado y para interconectar para enviar el paquete.

```
gila# show mpls forwarding-table Local Outgoing Prefix Bytes tag Outgoing Next Hop tag tag or VC
or Tunnel Id switched interface 16 Pop tag 88.1.1.0/24 0 Et1/1 88.1.2.2 Pop tag 88.1.1.0/24 0
Et1/0 88.1.3.2 17 Pop tag 88.1.4.0/24 0 Et1/1 88.1.2.2 18 Pop tag 88.1.10.0/24 0 Et1/1 88.1.2.2
19 Pop tag 88.1.11.1/32 0 Et1/1 88.1.2.2 20 Pop tag 88.1.5.0/24 0 Et1/0 88.1.3.2 21 19
88.1.11.10/32 0 Et1/1 88.1.2.2 22 88.1.11.10/32 0 Et1/0 88.1.3.2 22 20 172.18.60.176/32 0 Et1/1
88.1.2.2 23 172.18.60.176/32 0 Et1/0 88.1.3.2 23 Untagged 172.31.1.0/24[V] 6306 Fa0/0
10.88.162.6 24 Aggregate 10.88.162.4/30[V] 1920 25 Aggregate 10.88.162.8/30[V] 487120 26
Untagged 172.31.1.0/24[V] 1896 Et1/2 10.88.162.14 27 Aggregate 10.88.162.12/30[V] \ 972200 30
Pop tag 88.1.11.5/32 0 Et1/0 88.1.3.2 31 Pop tag 88.1.88.0/24 0 Et1/0 88.1.3.2 32 16
88.1.97.0/24 0 Et1/0 88.1.3.2 33 Pop tag 88.1.99.0/24 0 Et1/0 88.1.3.2 gila#
```

Configuraciones

Una cierta información extraña se ha quitado de las configuraciones para la brevedad.

```
IGUANA:
!
ip vrf custA
 rd 65002:100
 route-target export 65002:100
 route-target import 65002:100
!
ip vrf custB
 rd 65002:200
 route-target export 65002:200
 route-target import 65002:200
!
ip cef
mpls label protocol ldp
tag-switching tdp router-id Loopback0
!
interface Loopback0
 ip address 88.1.11.5 255.255.255.255
 no ip route-cache
 no ip mroute-cache
!
interface Loopback11
 ip vrf forwarding custA
 ip address 172.16.1.1 255.255.255.255
!
interface Ethernet1/0/0
 ip vrf forwarding custB
 ip address 10.88.163.5 255.255.255.252
 no ip route-cache
 no ip mroute-cache
!
interface Ethernet1/0/4
 ip address 88.1.1.1 255.255.255.0
 ip nat inside
 no ip mroute-cache
 tag-switching ip
!
interface Ethernet1/0/5
 ip address 88.1.3.2 255.255.255.0
 ip nat inside
 no ip mroute-cache
 tag-switching ip
!
!
interface FastEthernet1/1/0
 ip address 88.1.88.1 255.255.255.0
 ip nat outside
 full-duplex
!
interface FastEthernet5/0/0
 ip address 88.1.99.1 255.255.255.0
 speed 100
 full-duplex
!
router ospf 881
 log-adjacency-changes
 redistribute static subnets
 network 88.1.0.0 0.0.255.255 area 0
!
router bgp 65002
 no synchronization
 no bgp default ipv4-unicast
 bgp log-neighbor-changes
 neighbor 88.1.11.1 remote-as 65002
```

```
 neighbor 88.1.11.1 update-source Loopback0
 neighbor 88.1.11.9 remote-as 65002
 neighbor 88.1.11.9 update-source Loopback0
 neighbor 88.1.11.10 remote-as 65002
 neighbor 88.1.11.10 update-source Loopback0
 no auto-summary
 !
 address-family ipv4 multicast
 no auto-summary
 no synchronization
 exit-address-family
 !
 address-family vpnv4
 neighbor 88.1.11.1 activate
 neighbor 88.1.11.1 send-community extended
 neighbor 88.1.11.9 activate
 neighbor 88.1.11.9 send-community extended
 no auto-summary
 exit-address-family
 !
 address-family ipv4
 neighbor 88.1.11.1 activate
 neighbor 88.1.11.9 activate
 neighbor 88.1.11.10 activate
 no auto-summary
 no synchronization
 exit-address-family
 !
 address-family ipv4 vrf custB
 redistribute connected
 redistribute static
 no auto-summary
 no synchronization
 exit-address-family
 !
 address-family ipv4 vrf custA
 redistribute static
 no auto-summary
 no synchronization
 exit-address-family
!
ip nat pool SSPOOL1 192.168.1.1 192.168.1.254 prefix-length 24
ip nat inside source list 181 pool SSPOOL1 vrf custA overload
ip nat inside source list 181 pool SSPOOL1 vrf custB overload
ip classless
ip route 88.1.88.0 255.255.255.0 FastEthernet1/1/0
ip route 88.1.97.0 255.255.255.0 FastEthernet5/0/0 88.1.99.2
ip route 88.1.99.0 255.255.255.0 FastEthernet5/0/0 88.1.99.2
ip route 192.168.1.0 255.255.255.0 Null0
ip route vrf custA 88.1.88.8 255.255.255.255 FastEthernet1/1/0 88.1.88.8 global
ip route vrf custB 10.88.208.0 255.255.240.0 10.88.163.6
ip route vrf custB 64.102.0.0 255.255.0.0 10.88.163.6
ip route vrf custB 88.1.88.8 255.255.255.255 FastEthernet1/1/0 88.1.88.8 global
ip route vrf custB 128.0.0.0 255.0.0.0 10.88.163.6
no ip http server
!
access-list 181 permit ip any host 88.1.88.8
!
GILA:

!
ip vrf custA
 rd 65002:100
 route-target export 65002:100
```

```
 route-target import 65002:100
!
ip vrf custB
 rd 65002:200
 route-target export 65002:200
 route-target import 65002:200
!
ip cef
mpls label protocol ldp
tag-switching tdp router-id Loopback0
!
interface Loopback0
 ip address 88.1.11.9 255.255.255.255
!
interface FastEthernet0/0
 ip vrf forwarding custA
 ip address 10.88.162.5 255.255.255.252
 duplex full
!
interface Ethernet1/0
 ip address 88.1.3.1 255.255.255.0
 no ip mroute-cache
 duplex half
 tag-switching ip
!
interface Ethernet1/1
 ip address 88.1.2.1 255.255.255.0
 no ip mroute-cache
 duplex half
 tag-switching ip
!
interface Ethernet1/2
 ip vrf forwarding custB
 ip address 10.88.162.13 255.255.255.252
 ip ospf cost 100
 duplex half
!
interface FastEthernet2/0
 ip vrf forwarding custA
 ip address 10.88.162.9 255.255.255.252
 duplex full
!
router ospf 881
 log-adjacency-changes
 redistribute static subnets
 network 88.1.0.0 0.0.255.255 area 0
 default-metric 30
!
router bgp 65002
 no synchronization
 no bgp default ipv4-unicast
 bgp log-neighbor-changes
 neighbor 88.1.11.1 remote-as 65002
 neighbor 88.1.11.1 update-source Loopback0
 neighbor 88.1.11.1 activate
 neighbor 88.1.11.5 remote-as 65002
 neighbor 88.1.11.5 update-source Loopback0
 neighbor 88.1.11.5 activate
 no auto-summary
 !
 address-family ipv4 vrf custB
 redistribute connected
 redistribute static
 no auto-summary
```

```
 no synchronization
 exit-address-family
 !
 address-family ipv4 vrf custA
 redistribute connected
 redistribute static
 no auto-summary
 no synchronization
 exit-address-family
 !
 address-family vpnv4
 neighbor 88.1.11.1 activate
 neighbor 88.1.11.1 send-community extended
 neighbor 88.1.11.5 activate
 neighbor 88.1.11.5 send-community extended
 no auto-summary
 exit-address-family
!
ip classless
ip route vrf custA 172.31.1.0 255.255.255.0 FastEthernet0/0 10.88.162.6
ip route vrf custB 172.31.1.0 255.255.255.0 Ethernet1/2 10.88.162.14
!
```

El dragón del router tendría una configuración muy similar al **Gila**.

Importación/exportación de las blancos de la ruta no permitidas

Cuando la red de servicio compartida se configura como caso sí mismo VRF, el NAT central en la salida PE no es posible. Esto es porque los paquetes entrantes no pueden ser distinguidos y solamente una ruta de nuevo a la subred que origina está presente en la salida PE NAT.

Nota: Las visualizaciones que siguen se significan para ilustrar el resultado de una configuración no válida.

La red de muestra fue configurada de modo que la red de servicio compartida fuera definida como caso VRF (nombre VRF = sserver). Ahora, una visualización de la tabla CEF en el ingreso PE muestra esto:

```
gila# show ip cef vrf custA 88.1.88.0 88.1.88.0/24, version 45, epoch 0, cached adjacency
88.1.3.2 0 packets, 0 bytes tag information set local tag: VPN-route-head fast tag rewrite with
Et1/0, 88.1.3.2, tags imposed: {24} via 88.1.11.5, 0 dependencies, recursive next hop 88.1.3.2,
Ethernet1/0 via 88.1.11.5/32 valid cached adjacency tag rewrite with Et1/0, 88.1.3.2, tags
imposed: {24} gila# gila# show ip cef vrf custB 88.1.88.0 88.1.88.0/24, version 71, epoch 0,
cached adjacency 88.1.3.2 0 packets, 0 bytes tag information set local tag: VPN-route-head fast
tag rewrite with Et1/0, 88.1.3.2, tags imposed: {24} via 88.1.11.5, 0 dependencies, recursive
next hop 88.1.3.2, Ethernet1/0 via 88.1.11.5/32 valid cached adjacency tag rewrite with Et1/0,
88.1.3.2, tags imposed: {24} gila# iguana# show tag-switching forwarding vrftags 24 Local
Outgoing Prefix Bytes tag Outgoing Next Hop tag tag or VC or Tunnel Id switched interface 24
Aggregate 88.1.88.0/24[V] 10988 iguana#
```

Nota: Aviso cómo el valor *24 de la* etiqueta se impone para el custA VRF y el custB VRF.

Esta visualización muestra la tabla de ruteo para el caso compartido “sserver” del servicio VRF:

```
iguana# show ip route vrf sserver 172.31.1.1 Routing entry for 172.31.1.0/24 Known via "bgp
65002", distance 200, metric 0, type internal Last update from 88.1.11.9 1d01h ago Routing
Descriptor Blocks: * 88.1.11.9 (Default-IP-Routing-Table), from 88.1.11.9, 1d01h ago Route
metric is 0, traffic share count is 1 AS Hops 0
```

Nota: Solamente una ruta está presente para la red de destino de la perspectiva del router de la salida PE (**iguana**).

Por lo tanto, el tráfico del cliente múltiple VPN no podría ser distinguido y el tráfico de retorno no podría alcanzar el VPN apropiado. **En el caso donde el servicio compartido se debe definir como caso VRF, la función NAT se debe mover al ingreso PE.**

Ingreso PE NAT

En este ejemplo, el **Gila** y el **dragón** marcados los routeres de borde del proveedor se configuran para el NAT. Definen a un agrupamiento NAT para cada cliente asociado VPN que necesite el acceso al servicio compartido. El pool apropiado se utiliza para cada uno de los direccionamientos de red del cliente que son NATed. El NAT se realiza solamente en los paquetes destinados para el host compartido del servicio en 88.1.88.8.

```
ip nat pool SSPOOL1 192.168.1.1 192.168.1.254 prefix-length 24 ip nat pool SSPOOL2 192.168.2.1
192.168.2.254 prefix-length 24 ip nat inside source list 181 pool SSPOOL1 vrf custA overload ip
nat inside source list 181 pool SSPOOL2 vrf custB overload
```

Nota: En este escenario, no soportan a los pools compartidos. Si el servicio compartido LAN (en la salida PE) está conectado a través de una interfaz genérica, después el agrupamiento NAT puede ser compartido.

Un ping originado de una dirección duplicada (172.31.1.1) dentro de cada uno de las redes asoció al **neuse** y a los resultados **capefear8** en estas entradas de NAT:

Del Gila:

```
gila# show ip nat translations Pro Inside global Inside local Outside local Outside global icmp
192.168.1.1:2139 172.31.1.1:2139 88.1.88.8:2139 88.1.88.8:2139 icmp 192.168.1.1:2140
172.31.1.1:2140 88.1.88.8:2140 88.1.88.8:2140 icmp 192.168.1.1:2141 172.31.1.1:2141
88.1.88.8:2141 88.1.88.8:2141 icmp 192.168.1.1:2142 172.31.1.1:2142 88.1.88.8:2142
88.1.88.8:2142 icmp 192.168.1.1:2143 172.31.1.1:2143 88.1.88.8:2143 88.1.88.8:2143 icmp
192.168.2.2:676 172.31.1.1:676 88.1.88.8:676 88.1.88.8:676 icmp 192.168.2.2:677 172.31.1.1:677
88.1.88.8:677 88.1.88.8:677 icmp 192.168.2.2:678 172.31.1.1:678 88.1.88.8:678 88.1.88.8:678 icmp
192.168.2.2:679 172.31.1.1:679 88.1.88.8:679 88.1.88.8:679 icmp 192.168.2.2:680 172.31.1.1:680
88.1.88.8:680 88.1.88.8:680
```

Nota: Traducen a la misma dirección local interna (172.31.1.1) a cada uno de las agrupaciones definidas según la fuente VRF. El VRF se puede ver en el **comando verbose nacional de la traducción del IP de la demostración**:

```
gila# show ip nat translations verbose Pro Inside global Inside local Outside local Outside
global icmp 192.168.1.1:2139 172.31.1.1:2139 88.1.88.8:2139 88.1.88.8:2139 create 00:00:08, use
00:00:08, left 00:00:51, Map-Id(In): 3, flags: extended, use_count: 0, VRF : custA icmp
192.168.1.1:2140 172.31.1.1:2140 88.1.88.8:2140 88.1.88.8:2140 create 00:00:08, use 00:00:08,
left 00:00:51, Map-Id(In): 3, flags: extended, use_count: 0, VRF : custA icmp 192.168.1.1:2141
172.31.1.1:2141 88.1.88.8:2141 88.1.88.8:2141 create 00:00:08, use 00:00:08, left 00:00:51, Map-
Id(In): 3, flags: extended, use_count: 0, VRF : custA icmp 192.168.1.1:2142 172.31.1.1:2142
88.1.88.8:2142 88.1.88.8:2142 create 00:00:08, use 00:00:08, left 00:00:51, Map-Id(In): 3,
flags: extended, use_count: 0, VRF : custA icmp 192.168.1.1:2143 172.31.1.1:2143 88.1.88.8:2143
88.1.88.8:2143 create 00:00:08, use 00:00:08, left 00:00:51, Map-Id(In): 3, flags: extended,
use_count: 0, VRF : custA icmp 192.168.2.2:676 172.31.1.1:676 88.1.88.8:676 88.1.88.8:676 create
00:00:10, use 00:00:10, left 00:00:49, Map-Id(In): 2, flags: extended, use_count: 0, VRF : custB
icmp 192.168.2.2:677 172.31.1.1:677 88.1.88.8:677 88.1.88.8:677 create 00:00:10, use 00:00:10,
left 00:00:49, Map-Id(In): 2, flags: extended, use_count: 0, VRF : custB icmp 192.168.2.2:678
172.31.1.1:678 88.1.88.8:678 88.1.88.8:678 create 00:00:10, use 00:00:10, left 00:00:49, Map-
Id(In): 2, flags: extended, use_count: 0, VRF : custB icmp 192.168.2.2:679 172.31.1.1:679
88.1.88.8:679 88.1.88.8:679 create 00:00:10, use 00:00:10, left 00:00:49, Map-Id(In): 2, flags:
extended, use_count: 0, VRF : custB icmp 192.168.2.2:680 172.31.1.1:680 88.1.88.8:680
88.1.88.8:680 create 00:00:10, use 00:00:10, left 00:00:49, Map-Id(In): 2, flags: extended,
use_count: 0, VRF : custB
```

Estas visualizaciones muestran la información de ruteo para cada uno de los VPN localmente asociados para el cliente A y el cliente B:

```
gila# show ip route vrf custA Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP D -
EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area N1 - OSPF NSSA external type 1, N2 -
OSPF NSSA external type 2 E1 - OSPF external type 1, E2 - OSPF external type 2 I - IS-IS, L1 -
IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area * - candidate default, U - per-user
static route, o - ODR P - periodic downloaded static route Gateway of last resort is 88.1.11.1
to network 0.0.0.0     172.18.0.0/32 is subnetted, 2 subnets
B       172.18.60.179 [200/0] via 88.1.11.1, 00:03:59
B       172.18.60.176 [200/0] via 88.1.11.1, 00:03:59
     172.31.0.0/24 is subnetted, 1 subnets
S       172.31.1.0 [1/0] via 10.88.162.6, FastEthernet0/0
     10.0.0.0/8 is variably subnetted, 5 subnets, 2 masks
B       10.88.0.0/20 [200/0] via 88.1.11.1, 00:03:59
B       10.88.32.0/20 [200/0] via 88.1.11.1, 00:03:59
C       10.88.162.4/30 is directly connected, FastEthernet0/0
C       10.88.162.8/30 is directly connected, FastEthernet2/0
B       10.88.161.8/30 [200/0] via 88.1.11.1, 00:04:00
     88.0.0.0/24 is subnetted, 2 subnets
B       88.1.88.0 [200/0] via 88.1.11.5, 00:04:00
B       88.1.99.0 [200/0] via 88.1.11.5, 00:04:00
S 192.168.1.0/24 is directly connected, Null0 B* 0.0.0.0/0 [200/0] via 88.1.11.1, 00:04:00 gila#
show ip route vrf custB Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP D -
EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area N1 - OSPF NSSA external type 1, N2 -
OSPF NSSA external type 2 E1 - OSPF external type 1, E2 - OSPF external type 2 I - IS-IS, L1 -
IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area * - candidate default, U - per-user
static route, o - ODR P - periodic downloaded static route Gateway of last resort is not set
64.0.0.0/16 is subnetted, 1 subnets
B       64.102.0.0 [200/0] via 88.1.11.5, 1d21h
     172.18.0.0/32 is subnetted, 2 subnets
B       172.18.60.179 [200/0] via 88.1.11.1, 1d21h
B       172.18.60.176 [200/0] via 88.1.11.1, 1d21h
     172.31.0.0/24 is subnetted, 1 subnets
S       172.31.1.0 [1/0] via 10.88.162.14, Ethernet1/2
     10.0.0.0/8 is variably subnetted, 6 subnets, 3 masks
B       10.88.194.16/28 [200/100] via 88.1.11.1, 1d20h
B       10.88.208.0/20 [200/0] via 88.1.11.5, 1d21h
B       10.88.194.4/30 [200/100] via 88.1.11.1, 1d20h
B       10.88.163.4/30 [200/0] via 88.1.11.5, 1d21h
B       10.88.161.4/30 [200/0] via 88.1.11.1, 1d21h
C       10.88.162.12/30 is directly connected, Ethernet1/2
     11.0.0.0/24 is subnetted, 1 subnets
B       11.1.1.0 [200/100] via 88.1.11.1, 1d20h
     88.0.0.0/24 is subnetted, 2 subnets
B       88.1.88.0 [200/0] via 88.1.11.5, 1d21h
B       88.1.99.0 [200/0] via 88.1.11.5, 1d21h
S    192.168.2.0/24 is directly connected, Null0 B 128.0.0.0/8 [200/0] via 88.1.11.5, 1d21h
```

Nota: Una ruta para cada uno de los agrupamientos NAT se ha agregado de la configuración estática. Estas subredes se importan posteriormente en el servidor compartido VRF en la **iguana del** router de la salida PE:

```
iguana# show ip route vrf sserver Routing Table: sserver
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       I - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route
Gateway of last resort is not set     64.0.0.0/16 is subnetted, 1 subnets
B       64.102.0.0 [20/0] via 10.88.163.6 (custB), 1d20h
```

```
     172.18.0.0/32 is subnetted, 2 subnets
B       172.18.60.179 [200/0] via 88.1.11.1, 1d20h
B       172.18.60.176 [200/0] via 88.1.11.1, 1d20h
     172.31.0.0/24 is subnetted, 1 subnets
B       172.31.1.0 [200/0] via 88.1.11.9, 1d05h
     10.0.0.0/8 is variably subnetted, 8 subnets, 3 masks
B       10.88.194.16/28 [200/100] via 88.1.11.1, 1d20h
B       10.88.208.0/20 [20/0] via 10.88.163.6 (custB), 1d20h
B       10.88.194.4/30 [200/100] via 88.1.11.1, 1d20h
B       10.88.162.4/30 [200/0] via 88.1.11.9, 1d20h
B       10.88.163.4/30 is directly connected, 1d20h, Ethernet1/0/0
B       10.88.161.4/30 [200/0] via 88.1.11.1, 1d20h
B       10.88.162.8/30 [200/0] via 88.1.11.9, 1d20h
B       10.88.162.12/30 [200/0] via 88.1.11.9, 1d20h
     11.0.0.0/24 is subnetted, 1 subnets
B       11.1.1.0 [200/100] via 88.1.11.1, 1d20h
     12.0.0.0/24 is subnetted, 1 subnets
S       12.12.12.0 [1/0] via 88.1.99.10
     88.0.0.0/24 is subnetted, 3 subnets
C       88.1.88.0 is directly connected, FastEthernet1/1/0
S       88.1.97.0 [1/0] via 88.1.99.10
C       88.1.99.0 is directly connected, FastEthernet5/0/0
```

B 192.168.1.0/24 [200/0] via 88.1.11.9, 1d20h B 192.168.2.0/24 [200/0] via 88.1.11.9, 01:59:23 B 128.0.0.0/8 [20/0] via 10.88.163.6 (custB), 1d20h

Configuraciones

Una cierta información extraña se ha quitado de las configuraciones para la brevedad.

```
GILA:
ip vrf custA
 rd 65002:100
 route-target export 65002:100
 route-target export 65002:1001
 route-target import 65002:100
!
ip vrf custB
 rd 65002:200
 route-target export 65002:200
 route-target export 65002:2001
 route-target import 65002:200
 route-target import 65002:10
!
ip cef
mpls label protocol ldp
!interface Loopback0
 ip address 88.1.11.9 255.255.255.255
!
interface FastEthernet0/0
```

ip vrf forwarding custA ip address 10.88.162.5 255.255.255.252 **ip nat inside** duplex full ! interface Ethernet1/0 ip address 88.1.3.1 255.255.255.0 **ip nat outside** no ip mroute-cache duplex half tag-switching ip ! interface Ethernet1/1 ip address 88.1.2.1 255.255.255.0 **ip nat outside** no ip mroute-cache duplex half tag-switching ip ! interface Ethernet1/2 **ip vrf forwarding custB** ip address 10.88.162.13 255.255.255.252 **ip nat inside** duplex half ! router ospf 881 log-adjacency-changes redistribute static subnets network 88.1.0.0 0.0.255.255 area 0 default-metric 30 ! router bgp 65002 no synchronization no bgp default ipv4-unicast bgp log-neighbor-changes neighbor 88.1.11.1 remote-as 65002 neighbor 88.1.11.1 update-source Loopback0 neighbor 88.1.11.1 activate neighbor 88.1.11.5 remote-as 65002 neighbor 88.1.11.5 update-source Loopback0 neighbor 88.1.11.5 activate no auto-summary ! address-family ipv4 vrf custB redistribute connected redistribute static no auto-summary no synchronization exit-address-family ! address-family ipv4 vrf custA redistribute connected redistribute static no auto-summary no synchronization exit-address-family ! address-family vpnv4 neighbor 88.1.11.1 activate neighbor 88.1.11.1 send-community extended neighbor 88.1.11.5 activate neighbor 88.1.11.5 send-community extended no auto-summary exit-address-family ! ip nat pool **SSPOOL1** 192.168.1.1 192.168.1.254 prefix-length

```
24 ip nat pool SSPOOL2 192.168.2.1 192.168.2.254 prefix-length 24 ip nat inside source list 181
pool SSPOOL1 vrf custA overload ip nat inside source list 181 pool SSPOOL2 vrf custB overload ip
classless ip route vrf custA 172.31.1.0 255.255.255.0 FastEthernet0/0 10.88.162.6 ip route vrf
custA 192.168.1.0 255.255.255.0 Null0 ip route vrf custB 172.31.1.0 255.255.255.0 Ethernet1/2
10.88.162.14 ip route vrf custB 192.168.2.0 255.255.255.0 Null0 ! access-list 181 permit ip any
host 88.1.88.8 !
```

Nota: Las interfaces que hacen frente a las redes del cliente se señalan como las interfaces del "interior" NAT y las interfaces MPLS se señalan como NAT "exterior" interconectan.

```
iguana:
ip vrf custB
 rd 65002:200
 route-target export 65002:200
 route-target export 65002:2001
 route-target import 65002:200
 route-target import 65002:10
!
ip vrf sserver
 rd 65002:10
 route-target export 65002:10
 route-target import 65002:2001
 route-target import 65002:1001
!
ip cef distributed
mpls label protocol ldp
!interface Loopback0
 ip address 88.1.11.5 255.255.255.255
 no ip route-cache
 no ip mroute-cache
!
interface Ethernet1/0/0
 ip vrf forwarding custB
 ip address 10.88.163.5 255.255.255.252
 no ip route-cache
 no ip mroute-cache
!
interface Ethernet1/0/4
 ip address 88.1.1.1 255.255.255.0
 no ip route-cache
 no ip mroute-cache
 tag-switching ip
!
interface Ethernet1/0/5
 ip address 88.1.3.2 255.255.255.0
 no ip route-cache
 no ip mroute-cache
 tag-switching ip
!
interface FastEthernet1/1/0
 ip vrf forwarding sserver
 ip address 88.1.88.1 255.255.255.0
 no ip route-cache
 no ip mroute-cache
 full-duplex
!
router ospf 881
 log-adjacency-changes
 redistribute static subnets
 network 88.1.0.0 0.0.255.255 area 0
!
router bgp 65002
 no synchronization
 no bgp default ipv4-unicast
```

```
bgp log-neighbor-changes
neighbor 88.1.11.1 remote-as 65002
neighbor 88.1.11.1 update-source Loopback0
neighbor 88.1.11.9 remote-as 65002
neighbor 88.1.11.9 update-source Loopback0
neighbor 88.1.11.10 remote-as 65002
neighbor 88.1.11.10 update-source Loopback0
no auto-summary
!
address-family ipv4 multicast
no auto-summary
no synchronization
exit-address-family
!
address-family vpnv4
neighbor 88.1.11.1 activate
neighbor 88.1.11.1 send-community extended
neighbor 88.1.11.9 activate
neighbor 88.1.11.9 send-community extended
no auto-summary
exit-address-family
!
address-family ipv4
neighbor 88.1.11.1 activate
neighbor 88.1.11.9 activate
neighbor 88.1.11.10 activate
no auto-summary
no synchronization
exit-address-family
!
address-family ipv4 vrf sserver
redistribute connected
no auto-summary
no synchronization
exit-address-family
!
address-family ipv4 vrf custB
redistribute connected
redistribute static
no auto-summary
no synchronization
exit-address-family
```

El dragón del router tendría una configuración muy similar al **Gila**.

Paquetes que llegan el PE central después del ingreso PE NAT

Las trazas abajo ilustran el requisito para los agrupamientos NAT únicos cuando la red de servicio compartida destino se configura como caso VRF. Una vez más refiera al diagrama en el cuadro 5. Los paquetes mostrados abajo fueron capturados mientras que ingresaron la interfaz IP e1/0/5 MPLS en el **iguana del** router.

Generación de eco del cliente A VPN

Aquí, vemos un pedido de eco el venir de la dirección IP de origen 172.31.1.1 en el custA VRF. Han traducido a la dirección de origen a 192.168.1.1 según lo especificado por la configuración del NAT:

```
ip nat pool SSPOOL1 192.168.1.1 192.168.1.254 prefix-length 24
ip nat inside source list 181 pool SSPOOL1 vrf custA overload
```

```
DLC:  ----- DLC Header -----
        DLC:
        DLC:  Frame 1 arrived at  09:15:29.8157; frame size is 118 (0076 hex)
            bytes.
        DLC:  Destination = Station 005054D92A25
        DLC:  Source      = Station 0090BF9C6C1C
        DLC:  Ethertype   = 8847 (MPLS)
        DLC:
  MPLS: ----- MPLS Label Stack  -----
        MPLS:
        MPLS: Label Value = 00019 MPLS: Reserved For Experimental Use = 0 MPLS: Stack Value = 1
(Bottom of Stack) MPLS: Time to Live = 254 (hops) MPLS: IP: ----- IP Header ----- IP: IP:
Version = 4, header length = 20 bytes IP: Type of service = 00 IP: 000. .... = routine IP: ...0
.... = normal delay IP: .... 0... = normal throughput IP: .... .0.. = normal reliability IP:
.... ..0. = ECT bit - transport protocol will ignore the CE bit IP: .... ...0 = CE bit - no
congestion IP: Total length = 100 bytes IP: Identification = 0 IP: Flags = 0X IP: .0.. .... =
may fragment IP: ..0. .... = last fragment IP: Fragment offset = 0 bytes IP: Time to live = 254
seconds/hops IP: Protocol = 1 (ICMP) IP: Header checksum = 4AE6 (correct) IP: Source address =
[192.168.1.1] IP: Destination address = [88.1.88.8] IP: No options IP: ICMP: ----- ICMP header -
---- ICMP: ICMP: Type = 8 (Echo) ICMP: Code = 0 ICMP: Checksum = 932D (correct) ICMP: Identifier
= 3046 ICMP: Sequence number = 3245 ICMP: [72 bytes of data] ICMP: ICMP: [Normal end of "ICMP
header".] ICMP:
```

Generación de eco del cliente B VPN

Aquí, vemos un pedido de eco el venir de la dirección IP de origen 172.31.1.1 en el custB VRF. Han traducido a la dirección de origen a 192.168.2.1 según lo especificado por la configuración del NAT:

```
ip nat pool SSPOOL2 192.168.2.1 192.168.2.254 prefix-length 24
ip nat inside source list 181 pool SSPOOL2 vrf custB overload
DLC:  ----- DLC Header -----
        DLC:
        DLC:  Frame 11 arrived at  09:15:49.6623; frame size is 118 (0076 hex)
            bytes.
        DLC:  Destination = Station 005054D92A25
        DLC:  Source      = Station 0090BF9C6C1C
        DLC:  Ethertype   = 8847 (MPLS)
        DLC:
  MPLS: ----- MPLS Label Stack  -----
        MPLS:
        MPLS: Label Value = 00019 MPLS: Reserved For Experimental Use = 0 MPLS: Stack Value = 1
(Bottom of Stack) MPLS: Time to Live = 254 (hops) MPLS: IP: ----- IP Header ----- IP: IP:
Version = 4, header length = 20 bytes IP: Type of service = 00 IP: 000. .... = routine IP: ...0
.... = normal delay IP: .... 0... = normal throughput IP: .... .0.. = normal reliability IP:
.... ..0. = ECT bit - transport protocol will ignore the CE bit IP: .... ...0 = CE bit - no
congestion IP: Total length = 100 bytes IP: Identification = 15 IP: Flags = 0X IP: .0.. .... =
may fragment IP: ..0. .... = last fragment IP: Fragment offset = 0 bytes IP: Time to live = 254
seconds/hops IP: Protocol = 1 (ICMP) IP: Header checksum = 49D6 (correct) IP: Source address =
[192.168.2.2] IP: Destination address = [88.1.88.8] IP: No options IP: ICMP: ----- ICMP header -
---- ICMP: ICMP: Type = 8 (Echo) ICMP: Code = 0 ICMP: Checksum = AB9A (correct) ICMP: Identifier
= 4173 ICMP: Sequence number = 4212 ICMP: [72 bytes of data] ICMP: ICMP: [Normal end of "ICMP
header".]
```

Nota: El valor de etiqueta MPLS es *0019* en ambos paquetes mostrados arriba.

Respuesta de eco al cliente A VPN

Después, vemos una Respuesta de eco el volver al IP Address de destino 192.168.1.1 en el custA VRF. La función del ingreso PE NAT traduce a la dirección destino a 172.31.1.1.

To VRF custA: DLC: ----- DLC Header ----- DLC: DLC: Frame 2 arrived at 09:15:29.8198; frame size is 118 (0076 hex) bytes. DLC: Destination = Station 0090BF9C6C1C DLC: Source = Station 005054D92A25 DLC: Ethertype = 8847 (MPLS) DLC: MPLS: ----- MPLS Label Stack ----- MPLS: **MPLS: Label Value = 0001A** MPLS: Reserved For Experimental Use = 0 MPLS: Stack Value = 1 (Bottom of Stack) MPLS: Time to Live = 254 (hops) MPLS: IP: ----- IP Header ----- IP: IP: Version = 4, header length = 20 bytes IP: Type of service = 00 IP: 000. .... = routine IP: ...0 .... = normal delay IP: .... 0... = normal throughput IP: .... .0.. = normal reliability IP: .... ..0. = ECT bit - transport protocol will ignore the CE bit IP: ....0 = CE bit - no congestion IP: Total length = 100 bytes IP: Identification = 18075 IP: Flags = 4X IP: .1.. .... = don't fragment IP: ..0. .... = last fragment IP: Fragment offset = 0 bytes IP: Time to live = 254 seconds/hops IP: Protocol = 1 (ICMP) IP: Header checksum = C44A (correct) IP: Source address = [88.1.88.8] **IP: Destination address = [192.168.1.1]** IP: No options IP: ICMP: ----- ICMP header ----- ICMP: ICMP: Type = 0 (Echo reply) ICMP: Code = 0 ICMP: Checksum = 9B2D (correct) ICMP: Identifier = 3046 ICMP: Sequence number = 3245 ICMP: [72 bytes of data] ICMP: ICMP: [Normal end of "ICMP header".] ICMP:

Respuesta de eco al cliente B VPN

Aquí, vemos una Respuesta de eco el volver al IP Address de destino 192.168.1.1 en el custB VRF. La función del ingreso PE NAT traduce a la dirección destino a 172.31.1.1.

To VRF custB: DLC: ----- DLC Header ----- DLC: DLC: Frame 12 arrived at 09:15:49.6635; frame size is 118 (0076 hex) bytes. DLC: Destination = Station 0090BF9C6C1C DLC: Source = Station 005054D92A25 DLC: Ethertype = 8847 (MPLS) DLC: MPLS: ----- MPLS Label Stack ----- MPLS: **MPLS: Label Value = 0001D** MPLS: Reserved For Experimental Use = 0 MPLS: Stack Value = 1 (Bottom of Stack) MPLS: Time to Live = 254 (hops) MPLS: IP: ----- IP Header ----- IP: IP: Version = 4, header length = 20 bytes IP: Type of service = 00 IP: 000. .... = routine IP: ...0 .... = normal delay IP: .... 0... = normal throughput IP: .... .0.. = normal reliability IP: .... ..0. = ECT bit - transport protocol will ignore the CE bit IP: ....0 = CE bit - no congestion IP: Total length = 100 bytes IP: Identification = 37925 IP: Flags = 4X IP: .1.. .... = don't fragment IP: ..0. .... = last fragment IP: Fragment offset = 0 bytes IP: Time to live = 254 seconds/hops IP: Protocol = 1 (ICMP) IP: Header checksum = 75BF (correct) IP: Source address = [88.1.88.8] **IP: Destination address = [192.168.2.2]** IP: No options IP: ICMP: ----- ICMP header ----- ICMP: ICMP: Type = 0 (Echo reply) ICMP: Code = 0 ICMP: Checksum = B39A (correct) ICMP: Identifier = 4173 ICMP: Sequence number = 4212 ICMP: [72 bytes of data] ICMP: ICMP: [Normal end of "ICMP header".]

Nota: En los paquetes de devolución, los valores de etiqueta MPLS son incluidos y diferencian: *001A* para el custA VRF y *001D* para el custB VRF.

La generación de eco del cliente un destino de VPN es una interfaz genérica

Este conjunto siguiente de los paquetes muestra la diferencia cuando la interfaz al servicio compartido LAN es una interfaz genérica y no una parte de un caso VRF. Aquí, la configuración se ha cambiado para utilizar un pool común para ambos VPN locales con los IP Addresses que solapaban.

```
ip nat pool SSPOOL1 192.168.1.1 192.168.1.254 prefix-length 24 ip nat inside source list 181
pool SSPOOL1 vrf custA overload ip nat inside source list 181 pool SSPOOL1 vrf custB overload
DLC:  ----- DLC Header -----
        DLC:
        DLC:  Frame 1 arrived at  09:39:19.6580; frame size is 118 (0076 hex)
            bytes.
        DLC:  Destination = Station 005054D92A25
        DLC:  Source      = Station 0090BF9C6C1C
        DLC:  Ethertype   = 8847 (MPLS)
        DLC:
  MPLS: ----- MPLS Label Stack  -----
        MPLS:
        MPLS: Label Value = 00019 MPLS: Reserved For Experimental Use = 0 MPLS: Stack Value = 1
```

```
(Bottom of Stack) MPLS: Time to Live = 254 (hops) MPLS: IP: ----- IP Header ----- IP: IP:
Version = 4, header length = 20 bytes IP: Type of service = 00 IP: 000. .... = routine IP: ...0
.... = normal delay IP: .... 0... = normal throughput IP: .... .0.. = normal reliability IP:
.... ..0. = ECT bit - transport protocol will ignore the CE bit IP: .... ...0 = CE bit - no
congestion IP: Total length = 100 bytes IP: Identification = 55 IP: Flags = 0X IP: .0.. .... =
may fragment IP: ..0. .... = last fragment IP: Fragment offset = 0 bytes IP: Time to live = 254
seconds/hops IP: Protocol = 1 (ICMP) IP: Header checksum = 4AAF (correct) IP: Source address =
[192.168.1.1] IP: Destination address = [88.1.88.8] IP: No options IP: ICMP: ----- ICMP header -
---- ICMP: ICMP: Type = 8 (Echo) ICMP: Code = 0 ICMP: Checksum = 0905 (correct) ICMP: Identifier
= 874 ICMP: Sequence number = 3727 ICMP: [72 bytes of data] ICMP: ICMP: [Normal end of "ICMP
header".]
```

La generación de eco del destino de VPN del cliente B es una interfaz genérica

Aquí, vemos un pedido de eco el venir de la dirección IP de origen 172.31.1.1 en el custB VRF. Tradujeron a la dirección de origen a 192.168.1.3 (del pool común SSPOOL1) según lo especificado por la configuración del NAT:

```
ip nat pool SSPOOL1 192.168.1.1 192.168.1.254 prefix-length 24 ip nat inside source list 181
pool SSPOOL1 vrf custA overload ip nat inside source list 181 pool SSPOOL1 vrf custB overload
DLC:  ----- DLC Header -----
        DLC:
        DLC:  Frame 11 arrived at  09:39:26.4971; frame size is 118 (0076 hex)
            bytes.
        DLC:  Destination = Station 005054D92A25
        DLC:  Source      = Station 0090BF9C6C1C
        DLC:  Ethertype   = 8847 (MPLS)
        DLC:
  MPLS: ----- MPLS Label Stack  -----
        MPLS:
        MPLS: Label Value = 0001F MPLS: Reserved For Experimental Use = 0 MPLS: Stack Value = 1
(Bottom of Stack) MPLS: Time to Live = 254 (hops) MPLS: IP: ----- IP Header ----- IP: IP:
Version = 4, header length = 20 bytes IP: Type of service = 00 IP: 000. .... = routine IP: ...0
.... = normal delay IP: .... 0... = normal throughput IP: .... .0.. = normal reliability IP:
.... ..0. = ECT bit - transport protocol will ignore the CE bit IP: .... ...0 = CE bit - no
congestion IP: Total length = 100 bytes IP: Identification = 75 IP: Flags = 0X IP: .0.. .... =
may fragment IP: ..0. .... = last fragment IP: Fragment offset = 0 bytes IP: Time to live = 254
seconds/hops IP: Protocol = 1 (ICMP) IP: Header checksum = 4A99 (correct) IP: Source address =
[192.168.1.3] IP: Destination address = [88.1.88.8] IP: No options IP: ICMP: ----- ICMP header -
---- ICMP: ICMP: Type = 8 (Echo) ICMP: Code = 0 ICMP: Checksum = 5783 (correct) ICMP: Identifier
= 4237 ICMP: Sequence number = 977 ICMP: [72 bytes of data] ICMP: ICMP: [Normal end of "ICMP
header".]
```

Nota: Cuando la interfaz en la salida PE es una interfaz genérica (no un caso VRF), las escrituras de la etiqueta impuestas son diferentes. En este caso, *0x19* y *0x1F*.

La Respuesta de eco al cliente un destino de VPN es una interfaz genérica

Después, vemos una Respuesta de eco el volver al IP Address de destino 192.168.1.1 en el custA VRF. La función del ingreso PE NAT traduce a la dirección destino a 172.31.1.1.

```
DLC:  ----- DLC Header -----
        DLC:
        DLC:  Frame 2 arrived at  09:39:19.6621; frame size is 114 (0072 hex)
            bytes.
        DLC:  Destination = Station 0090BF9C6C1C
        DLC:  Source      = Station 005054D92A25
        DLC:  Ethertype   = 0800 (IP)
        DLC:
  IP: ----- IP Header -----
        IP:
```

```
        IP: Version = 4, header length = 20 bytes
        IP: Type of service = 00
        IP:       000. ....   = routine
        IP:       ...0 .... = normal delay
        IP:       .... 0... = normal throughput
        IP:       .... .0.. = normal reliability
        IP:       .... ..0. = ECT bit - transport protocol will ignore the CE
            bit
        IP:       .... ...0 = CE bit - no congestion
        IP: Total length    = 100 bytes
        IP: Identification  = 54387
        IP: Flags           = 4X
        IP:       .1.. .... = don't fragment
        IP:       ..0. .... = last fragment
        IP: Fragment offset = 0 bytes
        IP: Time to live    = 254 seconds/hops
        IP: Protocol        = 1 (ICMP)
        IP: Header checksum = 3672 (correct)
        IP: Source address      = [88.1.88.8]
        IP: Destination address = [192.168.1.1] IP: No options IP: ICMP: ----- ICMP header -----
ICMP: ICMP: Type = 0 (Echo reply) ICMP: Code = 0 ICMP: Checksum = 1105 (correct) ICMP:
Identifier = 874 ICMP: Sequence number = 3727 ICMP: [72 bytes of data] ICMP: ICMP: [Normal end
of "ICMP header".]
```

La Respuesta de eco al destino de VPN del cliente B es una interfaz genérica

Aquí, vemos una Respuesta de eco el volver al IP Address de destino 192.168.1.3 en el custB VRF. La función del ingreso PE NAT traduce a la dirección destino a 172.31.1.1.

```
DLC:  ----- DLC Header -----
        DLC:
        DLC:  Frame 12 arrived at  09:39:26.4978; frame size is 114 (0072 hex)
            bytes.
        DLC:  Destination = Station 0090BF9C6C1C
        DLC:  Source      = Station 005054D92A25
        DLC:  Ethertype   = 0800 (IP)
        DLC:
  IP: ----- IP Header -----
        IP:
        IP: Version = 4, header length = 20 bytes
        IP: Type of service = 00
        IP:       000. ....   = routine
        IP:       ...0 .... = normal delay
        IP:       .... 0... = normal throughput
        IP:       .... .0.. = normal reliability
        IP:       .... ..0. = ECT bit - transport protocol will ignore the CE
            bit
        IP:       .... ...0 = CE bit - no congestion
        IP: Total length    = 100 bytes
        IP: Identification  = 61227
        IP: Flags           = 4X
        IP:       .1.. .... = don't fragment
        IP:       ..0. .... = last fragment
        IP: Fragment offset = 0 bytes
        IP: Time to live    = 254 seconds/hops
        IP: Protocol        = 1 (ICMP)
        IP: Header checksum = 1BB8 (correct)
        IP: Source address      = [88.1.88.8]
        IP: Destination address = [192.168.1.3] IP: No options IP: ICMP: ----- ICMP header -----
ICMP: ICMP: Type = 0 (Echo reply) ICMP: Code = 0 ICMP: Checksum = 5F83 (correct) ICMP:
Identifier = 4237 ICMP: Sequence number = 977 ICMP: [72 bytes of data] ICMP: ICMP: [Normal end
of "ICMP header".]
```

Nota: Puesto que las contestaciones se destinan a una dirección global, no se impone ningunas

escrituras de la etiqueta VRF.

Con la interfaz de la salida al segmento LAN compartido del servicio definido como interfaz genérica, se permite un pool común. Los ping dan lugar a estas entradas de NAT en el router el **Gila**:

```
gila# show ip nat translations Pro Inside global Inside local Outside local Outside global icmp
192.168.1.3:4237 172.31.1.1:4237 88.1.88.8:4237 88.1.88.8:4237 icmp 192.168.1.3:4238
172.31.1.1:4238 88.1.88.8:4238 88.1.88.8:4238 icmp 192.168.1.3:4239 172.31.1.1:4239
88.1.88.8:4239 88.1.88.8:4239 icmp 192.168.1.3:4240 172.31.1.1:4240 88.1.88.8:4240
88.1.88.8:4240 icmp 192.168.1.3:4241 172.31.1.1:4241 88.1.88.8:4241 88.1.88.8:4241 icmp
192.168.1.1:874 172.31.1.1:874 88.1.88.8:874 88.1.88.8:874 icmp 192.168.1.1:875 172.31.1.1:875
88.1.88.8:875 88.1.88.8:875 icmp 192.168.1.1:876 172.31.1.1:876 88.1.88.8:876 88.1.88.8:876 icmp
192.168.1.1:877 172.31.1.1:877 88.1.88.8:877 88.1.88.8:877 icmp 192.168.1.1:878 172.31.1.1:878
88.1.88.8:878 88.1.88.8:878 gila# gila# show ip nat tr ver
Pro Inside global      Inside local       Outside local      Outside global
icmp 192.168.1.3:4237  172.31.1.1:4237    88.1.88.8:4237     88.1.88.8:4237
    create 00:00:08, use 00:00:08, left 00:00:51, Map-Id(In): 2,
    flags:
extended, use_count: 0, VRF : custB icmp 192.168.1.3:4238 172.31.1.1:4238 88.1.88.8:4238
88.1.88.8:4238 create 00:00:08, use 00:00:08, left 00:00:51, Map-Id(In): 2, flags: extended,
use_count: 0, VRF : custB icmp 192.168.1.3:4239 172.31.1.1:4239 88.1.88.8:4239 88.1.88.8:4239
create 00:00:08, use 00:00:08, left 00:00:51, Map-Id(In): 2, flags: extended, use_count: 0, VRF
: custB icmp 192.168.1.3:4240 172.31.1.1:4240 88.1.88.8:4240 88.1.88.8:4240 create 00:00:08, use
00:00:08, left 00:00:51, Map-Id(In): 2, flags: extended, use_count: 0, VRF : custB icmp
192.168.1.3:4241 172.31.1.1:4241 88.1.88.8:4241 88.1.88.8:4241 create 00:00:08, use 00:00:08,
left 00:00:51, Map-Id(In): 2, flags: extended, use_count: 0, VRF : custB icmp 192.168.1.1:874
172.31.1.1:874 88.1.88.8:874 88.1.88.8:874 create 00:00:16, use 00:00:16, left 00:00:43, Map-
Id(In): 3, Pro Inside global Inside local Outside local Outside global flags: extended,
use_count: 0, VRF : custA icmp 192.168.1.1:875 172.31.1.1:875 88.1.88.8:875 88.1.88.8:875 create
00:00:18, use 00:00:18, left 00:00:41, Map-Id(In): 3, flags: extended, use_count: 0, VRF : custA
icmp 192.168.1.1:876 172.31.1.1:876 88.1.88.8:876 88.1.88.8:876 create 00:00:18, use 00:00:18,
left 00:00:41, Map-Id(In): 3, flags: extended, use_count: 0, VRF : custA icmp 192.168.1.1:877
172.31.1.1:877 88.1.88.8:877 88.1.88.8:877 create 00:00:18, use 00:00:18, left 00:00:41, Map-
Id(In): 3, flags: extended, use_count: 0, VRF : custA icmp 192.168.1.1:878 172.31.1.1:878
88.1.88.8:878 88.1.88.8:878 create 00:00:18, use 00:00:18, left 00:00:41, Map-Id(In): 3, flags:
extended, use_count: 0, VRF : custA gila# debug ip nat vrf IP NAT VRF debugging is on gila# .Jan
2 09:34:54 EST: NAT-TAGSW(p) : Tag Pkt s=172.18.60.179, d=10.88.162.9, vrf=custA .Jan 2 09:35:02
EST: NAT-TAGSW(p) : Tag Pkt s=172.18.60.179, d=10.88.162.13, vrf=custB .Jan 2 09:35:12 EST: NAT-
ip2tag : Tag Pkt s=172.31.1.1, d=88.1.88.8, vrf=custA .Jan 2 09:35:12 EST: NAT-ip2tag: Punting
to process .Jan 2 09:35:12 EST: NAT-ip2tag : Tag Pkt s=172.31.1.1, d=88.1.88.8, vrf=custA .Jan 2
09:35:12 EST: NAT-ip2tag: Punting to process .Jan 2 09:35:12 EST: NAT-ip2tag : Tag Pkt
s=172.31.1.1, d=88.1.88.8, vrf=custA .Jan 2 09:35:12 EST: NAT-ip2tag: Punting to process .Jan 2
09:35:12 EST: NAT-ip2tag : Tag Pkt s=172.31.1.1, d=88.1.88.8, vrf=custA .Jan 2 09:35:12 EST:
NAT-ip2tag: Punting to process .Jan 2 09:35:12 EST: NAT-ip2tag : Tag Pkt s=172.31.1.1,
d=88.1.88.8, vrf=custA .Jan 2 09:35:12 EST: NAT-ip2tag: Punting to process .Jan 2 09:35:19 EST:
NAT-ip2tag : Tag Pkt s=172.31.1.1, d=88.1.88.8, vrf=custB .Jan 2 09:35:19 EST: NAT-ip2tag:
Punting to process .Jan 2 09:35:19 EST: NAT-ip2tag : Tag Pkt s=172.31.1.1, d=88.1.88.8,
vrf=custB .Jan 2 09:35:19 EST: NAT-ip2tag: Punting to process .Jan 2 09:35:19 EST: NAT-ip2tag :
Tag Pkt s=172.31.1.1, d=88.1.88.8, vrf=custB .Jan 2 09:35:19 EST: NAT-ip2tag: Punting to process
.Jan 2 09:35:19 EST: NAT-ip2tag : Tag Pkt s=172.31.1.1, d=88.1.88.8, vrf=custB .Jan 2 09:35:19
EST: NAT-ip2tag: Punting to process .Jan 2 09:35:19 EST: NAT-ip2tag : Tag Pkt s=172.31.1.1,
d=88.1.88.8, vrf=custB .Jan 2 09:35:19 EST: NAT-ip2tag: Punting to process gila#
```

Mantenga el ejemplo

Un ejemplo IP virtual de un servicio compartido PBX se muestra en el cuadro 8. Esto ilustra una variante a los ejemplos del ingreso y de la salida descritos anterior.

En este diseño, un conjunto de routers delantero-termina al servicio de VoIP compartido que realiza la función NAT. Este Routers tiene interfaces múltiples VRF usando una característica

conocida como VRF-Lite. El tráfico entonces fluye al clúster del Cisco CallManager compartido. Proporcionan los servicios del Firewall también en una base de la por-compañía. Las llamadas entre compañías deben pasar con el Firewall, mientras que las llamadas de la intra-compañía se manejan a través del cliente VPN usando el esquema de direccionamiento interno de la compañía.

Figura 8: Ejemplo virtual manejado del servicio PBX

Disponibilidad

El soporte del Cisco IOS NAT para el MPLS VPNs está disponible en el Cisco IOS Release 12.2(13)T y está disponible para todas las Plataformas que soporten el MPLS y puedan funcionar con este tren de la versión de despliegue temprana.

Conclusión

El Cisco IOS NAT tiene características para permitir el despliegue scalable de los servicios compartidos hoy. Cisco continúa desarrollando el soporte del gateway del nivel de la aplicación NAT (ALG) para los protocolos importantes para los clientes. Las mejoras del rendimiento y la aceleración por hardware para las funciones de traducción se asegurarán de que el NAT y ALGs proporcionen las soluciones aceptables durante un tiempo. Todas las actividades de los estándares relevantes y acciones comunitarias están siendo monitoreadas por Cisco. Pues se desarrollan otros estándares, su uso será evaluado basó en los deseos, los requisitos, y la aplicación del cliente.

Información Relacionada

- Gatewayes de capa de aplicación del Cisco IOS NAT
- MPLS y arquitecturas de VPN
- Diseño avanzado y implementación MPLS
- Soporte Técnico y Documentación - Cisco Systems