Cisco IOS NAT - Integração com MPLS VPN

Índice

Introdução

O software do Network Address Translation (NAT) do ® do Cisco IOS permite o acesso aos serviços compartilhados do MPLS VPNs múltiplo, mesmo quando os dispositivos nos VPN usam os endereços IP de Um ou Mais Servidores Cisco ICM NT que sobrepõem. O Cisco IOS NAT é preparado para VRF e pode ser configurado em roteadores na extremidade do provedor dentro da rede MPLS.

Nota: O MPLS nos IO é apoiado somente com legado NAT. Neste tempo, não há nenhum apoio no Cisco IOS para NAT NVI com MPLS.

O desenvolvimento do MPLS VPNs é projetado aumentar rapidamente durante os próximos vários anos. Os benefícios de uma infraestrutura de rede comum que a expansão rápida das licenças e as opções de conectividade flexíveis conduzam indubitavelmente um crescimento mais adicional nos serviços que podem ser oferecidos à comunidade da rede interna.

Contudo, as barreiras ao crescimento ainda permanecem. O IPv6 e sua promessa de um espaço de endereços IP que exceda as necessidades da Conectividade para o futuro próximo realizam-se ainda nas fases adiantadas de desenvolvimento. Das redes existentes métodos de endereçamento do IP privado do uso geralmente como definidos dentro do RFC 1918 . A tradução de endereço de rede é usada frequentemente interconectar redes quando os espaços de endereços sobrepõem ou a duplicação existe.

Os provedores de serviços e as empresas que têm serviços que do aplicativo de rede querem oferecer ou a parte com clientes e Parceiros quererão minimizar toda a carga da Conectividade colocada no usuário do serviço. É desejável, mesmo imperativo, estender o oferecimento a tantos como usuários potenciais porque necessário conseguir os objetivos desejados ou retornar. O esquema de endereçamento de IP no uso não deve ser uma barreira que exclua usuários

potenciais.

Pelo Cisco IOS de distribuição NAT dentro da infraestrutura comum do MPLS VPN, os provedores de serviços das comunicações podem aliviar algum da carga da Conectividade em clientes e acelerar sua capacidade para ligar serviços de aplicativo mais compartilhados a mais consumidores daqueles serviços.

Benefícios do NAT – Integração de MPLS

A integração NAT com MPLS tem benefícios para ambos os provedores de serviços e seus clientes de empreendimento. Oferece a provedores de serviços mais opções distribuir serviços compartilhados e fornecer o acesso 2 aqueles serviços. As ofertas do serviço adicional podem ser um diferenciador sobre concorrentes.

Para o provedor de serviços	Para o VPN
Mais ofertas de serviço	Custos reduzidos
Opções aumentadas do acesso	Acesso mais simples
Rendimento aumentado	Endereçando a flexibilidade

Os clientes de empreendimento que procuram externalizar alguma de sua carga de trabalho atual podem igualmente tirar proveito de umas ofertas mais largas por provedores de serviços. Deslocar a carga de executar toda a tradução de endereços necessária à rede de provedor de serviços alivia-os de umas tarefas administrativas complicadas. Os clientes podem continuar a usar o endereçamento privado, contudo mantêm o acesso aos serviços compartilhados e ao Internet. Consolidar a função NAT dentro da rede de provedor de serviços pode igualmente abaixar os custos total aos clientes de empreendimento desde que os roteadores de ponta do cliente não têm que executar a função NAT.

Considerações do projeto

Ao considerar os projetos que invocarão o NAT dentro da rede MPLS, a primeira etapa é determinar as necessidades do serviço de um ponto de vista do aplicativo. Você precisará de considerar os protocolos comunicação usada e toda a especial do cliente/server imposta pelo aplicativo. Certifique-se de que o apoio necessário para os protocolos empregados está apoiado e segurado pelo Cisco IOS NAT. Uma lista de protocolos suportados é fornecida nas gateway de camadas de aplicativo do Cisco IOS NAT do documento.

Em seguida, será necessário determinar o uso previsto do serviço compartilhado e a taxa de tráfego antecipada no pacote-por-segunda. O NAT é uma função do processo intensivo de cpu do roteador. Consequentemente, os requisitos de desempenho serão um fator em selecionar uma opção de distribuição particular e determinarão o número de dispositivos NAT envolvidos.

Também, considere todas as questões de segurança e precauções que deverem ser tomadas. Embora o MPLS VPNs, por definição, seja privado e eficazmente o tráfego separado, a rede de serviço compartilhada seja geralmente comum entre muitos VPN.

Cenários de distribuição

Há duas opções para o desenvolvimento NAT dentro da ponta de provedor MPLS:

- Centralizado com saída NAT PE
- Distribuído com ingresso NAT PE

Algumas vantagens a configurar a função NAT no ponto de saída da rede MPLS o mais próximo à rede de serviço compartilhada incluem:

- Uma configuração centralizada que promova um abastecimento mais simples do serviço
- Troubleshooting simplificado
- Escalabilidade operacional aumentada
- Exigências diminuídas da alocação de endereço IP

Contudo, as vantagens são deslocadas por uma redução na escalabilidade e no desempenho. Esta é as trocas principais que devem ser consideradas. Naturalmente, a função NAT pode igualmente ser executada dentro das redes cliente se se determina que a integração desta característica com uma rede MPLS não é desejável.

Ingresso PE NAT

O NAT pode ser configurado no roteador de PE do ingresso da rede MPLS segundo as indicações de figura 1. Com este projeto, a escalabilidade está mantida em grande parte quando o desempenho for aperfeiçoado distribuindo a função NAT sobre muitos dispositivos de ponta. Cada NAT PE segura o tráfego para os locais conectados localmente a esse PE. Regras NAT e listas de controle de acesso ou controle dos mapas de rota que os pacotes exigem a tradução.

Figura 1: Ingresso PE NAT

Há uma limitação que impeça um NAT entre dois VRF ao igualmente fornecer o NAT a um serviço compartilhado segundo as indicações de figura 2. Isto é devido à exigência designar relações como da “parte externa” NAT relações do “interior” e. O apoio para conexões entre VRF em um único PE é planejado para um Cisco IOS Release futuro.

Figura 2: Interempresarial

Saída PE NAT

O NAT pode ser configurado no roteador de PE da saída da rede MPLS segundo as indicações de figura 3. Com este projeto, a escalabilidade é reduzida a algum grau desde que o PE central deve manter rotas para todas as redes cliente que alcançam o serviço compartilhado. As exigências de desempenho do aplicativo devem igualmente ser consideradas de modo que o tráfego não sobrecarregue o roteador que deve traduzir os endereços IP de Um ou Mais Servidores Cisco ICM NT dos pacotes. Porque o NAT ocorre centralmente para todos os clientes que usam este trajeto, as associações do endereço IP de Um ou Mais Servidores Cisco ICM NT podem ser compartilhadas; assim, o número total de sub-redes exigidas é reduzido.

Figura 3: Saída PE NAT

Os roteadores múltiplos podiam ser distribuídos para aumentar a escalabilidade do projeto da saída PE NAT segundo as indicações de figura 4. Nesta encenação, o cliente VPN poderia ser “fornecida” em um roteador NAT específico. A tradução de endereço de rede ocorreria para o tráfego agregado a e do serviço compartilhado para aquele ajustou-se dos VPN. Por exemplo, o tráfego dos VPN para o cliente A e B poderia usar o NAT-PE1, quando o tráfego a e do VPN para o C do cliente usar o NAT-PE2. Cada NAT PE levaria o tráfego somente para os VPN específicos definidos e manteria somente rotas de volta aos locais naqueles VPN. Os poois de endereço NAT

separados poderiam ser definidos dentro de cada um dos roteadores de PE NAT de modo que os pacotes fossem distribuídos da rede de serviço compartilhada ao NAT apropriado PE para a tradução e o roteamento de volta ao cliente VPN.

Figura 4: Saída múltipla PE NAT

O projeto centralizado impõe uma limitação em como a rede de serviço compartilhada deve ser configurada. Especificamente, o uso da importação/exportação de rotas do MPLS VPN entre um serviço compartilhado VPN e o cliente VPN não são possíveis. Isto é devido à natureza da operação MPLS como especificado pelo RFC 2547 . Quando as rotas são importadas e exportadas usando as comunidades extendida e os descritores de rota, o NAT não pode determinar a fonte VPN do pacote que entra o NAT central PE. O caso usual é fazer à rede de serviço compartilhada uma interface genérica um pouco do que uma relação VRF. Uma rota à rede de serviço compartilhada é adicionada então no NAT central PE para cada tabela VRF associada com um cliente VPN que precisa o acesso ao serviço compartilhado como parte do processo de provisionamento. Isto é descrito com maiores detalhes mais tarde.

Opções de distribuição e detalhes de configuração

Esta seção inclui alguns detalhes relativos a cada um das opções de distribuição. Todos os exemplos são tomados da rede mostrada na figura 5. Refira este diagrama para o resto desta seção.

Nota: Na rede usada para ilustrar a operação de VRF NAT para este papel, somente os roteadores de PE são incluídos. Não há nenhum Roteadores do núcleo “P”. Contudo, os mecanismos essenciais podem ainda ser considerados.

Figura 5: Exemplo da configuração de NAT VRF

Saída PE NAT

Neste exemplo, o **gila** e o **dragão** marcados roteadores de extremidade do provedor são configurados como roteadores de PE simples. O PE central perto do serviço compartilhado LAN (**iguana**) é configurado para o NAT. Um único conjunto NAT é compartilhado por cada cliente VPN que precisa o acesso ao serviço compartilhado. O NAT é executado somente nos pacotes destinados para o host compartilhado do serviço em 88.1.88.8.

Encaminhamento de dados da saída PE NAT

Com MPLS, cada pacote incorpora a rede em um ingresso PE e retira a rede MPLS em uma saída PE. O trajeto dos Label Switching Router atravessados do ingresso à saída é sabido como o caminho comutado por rótulo (LSP). O LSP é unidirecional. Um LSP diferente é usado para o tráfego de retorno.

Ao usar a saída PE NAT, um Forwarding Equivalence Class (FEC) é definido eficazmente para todo o tráfego dos usuários do serviço compartilhado. Ou seja todos os pacotes destinados para o serviço compartilhado LAN são membros de um FEC comum. Um pacote é atribuído a um FEC particular apenas uma vez na borda de ingresso da rede e segue o LSP à saída PE. O FEC é designado no pacote de dados adicionando uma etiqueta particular.

Fluxo de pacote de informação ao serviço compartilhado do VPN

Para que dispositivos nos VPN múltiplos que têm os esquemas do endereço de sobreposição

para alcançar um host compartilhado do serviço, o NAT é exigido. Quando o NAT é configurado na saída PE, as entradas de tabela da tradução de endereço de rede incluirão um identificador VRF para diferenciar endereços duplicados e assegurar o roteamento apropriado.

Figura 6: Pacotes transmitidos à saída PE NAT

A figura 6 ilustra os pacotes destinados para um host compartilhado do serviço dois do cliente VPN que têm métodos de endereçamento do IP duplicado. A figura mostra que um pacote que origina no cliente A com um endereço de origem de 172.31.1.1 destinou para um server compartilhado em 88.1.88.8. Um outro pacote do cliente B com o mesmo endereço IP de origem é enviado igualmente ao mesmo server compartilhado. Quando os pacotes alcançam o roteador de PE, uma consulta da camada 3 está feita para a rede do IP de destino no banco de informação de encaminhamento (FIB).

A entrada MENTIR diz o roteador de PE para enviar o tráfego à saída PE usando uma pilha de rótulo. O rótulo inferior na pilha é atribuído pelo roteador de PE do destino, neste caso **iguana do** roteador.

```
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#
```

Nós podemos ver do indicador que pacotes do custA VRF teremos um valor da etiqueta de 24 (0x18) e os pacotes do custB VRF terão um valor da etiqueta de 28 (0x1C).

Neste caso, porque não há nenhum Roteadores “P” em nossa rede, lá não está nenhuma etiqueta adicional imposta. Tinha havido roteadores centrais, uma etiqueta exterior seria imposta e o processo normal de troca da etiqueta ocorreu dentro da rede central até que o pacote alcançou a saída PE.

Desde que o roteador de **gila** é conectado diretamente à saída PE, nós vemos que a etiqueta está estalada antes que esteja adicionada 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#
```

Os indicadores seguintes descrevem pacotes de eco como recebidos pelo roteador NAT da saída PE (na relação E1/0/5 na **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".]

Estes sibilos conduzem às seguintes entradas que estão sendo criadas na tabela NAT na **iguana do** roteador de PE da saída. As entradas específicas criadas para os pacotes mostrados acima podem ser combinadas por seu 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#

Fluxo de pacote de informação do serviço compartilhado de volta à origem VPN

Enquanto os pacotes fluem de volta aos dispositivos que alcançaram o host compartilhado do

serviço, a tabela NAT é examinada antes do roteamento (pacotes que vão da relação da “parte externa” NAT à relação do “interior”). Porque cada entrada exclusiva inclui o identificador correspondente VRF, o pacote pode ser traduzido e distribuído apropriadamente.

Figura 7: Pacotes transmitidos de volta ao usuário de serviço compartilhado

Segundo as indicações da figura 7, o tráfego de retorno é examinado primeiramente pelo NAT para encontrar uma entrada de tradução de harmonização. Por exemplo, um pacote é enviado ao destino 192.168.1.1. A tabela NAT é procurada. Quando o fósforo é encontrado, a tradução apropriada está feita ao endereço do “Inside Local” (172.31.1.1) e uma consulta da adjacência é executada então usando o VRF associado ID da entrada 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#
```

A etiqueta 23 (0x17) é usada para o tráfego destinado para 172.31.1.0/24 no custA VRF e na etiqueta 26 (0x1A) é usada para os pacotes destinados para 172.31.1.0/24 no custB VRF.

Isto é visto nos pacotes de resposta de eco enviados da **iguana do** roteador:

```
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".]
```

Quando o pacote alcança o roteador de PE do destino, a etiqueta está usada para determinar o VRF e a relação apropriados enviar sobre o pacote.

```
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#
```

Configurações

Alguma informação estranha foi removida das configurações para a brevidade.

```
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
!
```

O dragão do roteador teria uma configuração muito similar a gila.

Importação/exportação dos alvos da rota não permitidos

Quando a rede de serviço compartilhada é configurada como um exemplo próprio VRF, o NAT central na saída PE não é possível. Isto é porque os pacotes recebidos não podem ser distintos e somente uma rota de volta à sub-rede de origem esta presente na saída PE NAT.

Nota: Os indicadores que seguem são significados ilustrar o resultado de uma configuração inválida.

O exemplo de rede foi configurado de modo que a rede de serviço compartilhada fosse definida como um exemplo VRF (nome VRF = sserver). Agora, um indicador da tabela de CEF no ingresso PE mostra este:

```
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: Observação como o valor *24 da* etiqueta é imposto para o custA VRF e o custB VRF.

Este indicador mostra a tabela de roteamento para o exemplo compartilhado “sserver” do serviço 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: Somente uma rota esta presente para a rede de destino da perspectiva do roteador de PE da saída (iguana).

Consequentemente, o tráfego do cliente múltiplo VPN não poderia ser distinto e o tráfego de retorno não poderia alcançar o VPN apropriado. **No caso onde o serviço compartilhado deve ser definido como um exemplo VRF, a função NAT deve ser movida para o ingresso PE.**

Ingresso PE NAT

Neste exemplo, o **gila** e o **dragão** marcados roteadores de extremidade do provedor são configurados para o NAT. Um conjunto NAT é definido para cada cliente anexado VPN que precisa o acesso ao serviço compartilhado. O pool apropriado é usado para cada um dos endereços de rede cliente que são NATed. O NAT é executado somente nos pacotes destinados para o host compartilhado do serviço em 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: Nesta encenação, as associações compartilhadas não são apoiadas. Se o serviço compartilhado LAN (na saída PE) é conectado através de uma interface genérica, a seguir o conjunto NAT pode ser compartilhado.

Um sibilo originado de um endereço duplicado (172.31.1.1) dentro de cada um das redes anexou ao **neuse** e aos resultados **capefear8 n**estas entradas NAT:

De 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: O mesmo endereço local interno (172.31.1.1) é traduzido a cada um dos conjuntos definidos de acordo com a fonte VRF. O VRF pode ser visto no **comando verbose nat da tradução da mostra IP**:

```
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
```

Estes indicadores mostram a informação de roteamento para cada um dos VPN localmente anexados para o cliente A e o 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: Uma rota para cada um dos conjuntos NAT foi adicionada da configuração estática. Estas sub-redes são importadas subseqüentemente no server compartilhado VRF na iguana do roteador de PE da saída:

```
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

Configurações

Alguma informação estranha foi removida das configurações para a brevidade.

```
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: As relações que enfrentam as redes cliente são designadas como relações do “interior” NAT e as relações MPLS são designadas como NAT “parte externa” conectam.

```
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
```

O dragão do roteador teria uma configuração muito similar a gila.

Pacotes que chegam no PE central após o ingresso PE NAT

Os traços abaixo ilustram a exigência para conjuntos NAT originais quando a rede de serviço compartilhada destino é configurada como um exemplo VRF. Além disso, refira o diagrama na figura 5. Os pacotes mostrados abaixo foram capturados enquanto incorporaram a interface IP e1/0/5 MPLS na iguana do roteador.

Eco do cliente A VPN

Aqui, nós vemos uma requisição de eco vir do endereço IP de origem 172.31.1.1 no custA VRF. O endereço de origem foi traduzido a 192.168.1.1 como especificado pela configuração de 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:
```

Eco do cliente B VPN

Aqui, nós vemos uma requisição de eco vir do endereço IP de origem 172.31.1.1 no custB VRF. O endereço de origem foi traduzido a 192.168.2.1 como especificado pela configuração de 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: O valor de rótulo MPLS é *0019 em* ambos os pacotes mostrados acima.

Resposta de eco ao cliente A VPN

Em seguida, nós vemos uma resposta de eco ir para trás ao endereço IP de destino 192.168.1.1 no custA VRF. O endereço de destino é traduzido a 172.31.1.1 pela função do ingresso PE NAT.

```
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:

Resposta de eco ao cliente B VPN

Aqui, nós vemos uma resposta de eco ir para trás ao endereço IP de destino 192.168.1.1 no custB VRF. O endereço de destino é traduzido a 172.31.1.1 pela função do ingresso PE NAT.

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: Nos pacotes de informação de retorno, os valores de rótulo MPLS são incluídos e diferem: *001A* para o custA VRF e *001D* para o custB VRF.

O eco do cliente um destino de VPN é uma interface genérica

Este grupo seguinte de pacotes mostra a diferença quando a relação ao serviço compartilhado LAN é uma interface genérica e não parte de um exemplo VRF. Aqui, a configuração foi mudada para usar um pool comum para ambos os VPN locais com endereços IP de Um ou Mais Servidores Cisco ICM NT de sobreposição.

```
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".]
```

O eco do destino de VPN do cliente B é uma interface genérica

Aqui, nós vemos uma requisição de eco vir do endereço IP de origem 172.31.1.1 no custB VRF. O endereço de origem foi traduzido a 192.168.1.3 (do pool comum SSPOOL1) como especificado pela configuração de 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: Quando a relação na saída PE for uma interface genérica (não um exemplo VRF), as etiquetas impostas são diferentes. Neste caso, *0x19* e *0x1F*.

A resposta de eco ao cliente um destino de VPN é uma interface genérica

Em seguida, nós vemos uma resposta de eco ir para trás ao endereço IP de destino 192.168.1.1 no custA VRF. O endereço de destino é traduzido a 172.31.1.1 pela função do ingresso PE NAT.

```
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".]
```

A resposta de eco ao destino de VPN do cliente B é uma interface genérica

Aqui, nós vemos uma resposta de eco ir para trás ao endereço IP de destino 192.168.1.3 no custB VRF. O endereço de destino é traduzido a 172.31.1.1 pela função do ingresso PE NAT.

```
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: Desde que as respostas são destinadas a um endereço global, nenhuma etiqueta VRF é imposta.

Com a relação da saída ao segmento de LAN compartilhado do serviço definido como uma interface genérica, um pool comum é permitido. Os sibilos conduzem a estas entradas NAT no roteador **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#
```

Preste serviços de manutenção ao exemplo

Um exemplo de um serviço compartilhado do IP virtual PBX é mostrado em figura 8. Isto ilustra uma variação aos exemplos do ingresso e da saída descritos mais cedo.

Neste projeto, o serviço voip compartilhado dianteiro-é terminado por um conjunto de roteador que executa a função NAT. Este Roteadores tem relações múltiplas VRF usando uma característica conhecida como VRF-Lite. O tráfego flui então ao Cluster do CallManager daCisco compartilhado. Os serviços de firewall são proporcionados igualmente em uma base da por-

empresa. os atendimentos da Inter-empresa devem passar com o Firewall, quando os atendimentos intra-empresa forem segurados através do cliente VPN usando o método de endereçamento interno da empresa.

Figura 8: Exemplo virtual controlado do serviço PBX

Disponibilidade

O apoio do Cisco IOS NAT para o MPLS VPNs está disponível no Cisco IOS Release 12.2(13)T e está disponível para todas as Plataformas que apoiam o MPLS e podem executar este trem da versão de distribuição precoce.

Conclusão

O Cisco IOS NAT tem as características para permitir hoje o desenvolvimento escalável de serviços compartilhados. Cisco continua a desenvolver o apoio do gateway do nível do aplicativo NAT (ALG) para os protocolos importantes para clientes. As melhorias de desempenho e a aceleração de hardware para funções de tradução assegurar-se-ão de que o NAT e ALGs forneçam soluções aceitáveis por algum tempo para vir. Todas as atividades dos padrões relevantes e ações comunitárias estão sendo monitoradas por Cisco. Porque outros padrões são desenvolvidos, seu uso será avaliado baseou em desejos, em exigências, e em aplicativo do cliente.

Informações Relacionadas

- Gateway de camadas de aplicativo do Cisco IOS NAT
- MPLS e arquiteturas VPN
- Projeto avançado e aplicação MPLS
- Suporte Técnico e Documentação - Cisco Systems