ПРЕОБРАЗОВАНИЕ СЕТЕВЫХ АДРЕСОВ В CISCO IOS - интеграция с MPLS VPN

Содержание

Введение

Программное обеспечение IOS® Network Address Translation (NAT) Cisco предоставляет доступ к совместно используемым сервисам от множественных MPLS VPN, даже когда устройства в VPN используют IP-адреса то наложение. ПРЕОБРАЗОВАНИЕ СЕТЕВЫХ АДРЕСОВ В CISCO IOS осведомлено о VRF и может быть настроено на маршрутизаторах на стороне провайдера в сети MPLS.

Примечание: MPLS в IOS поддерживается только с устаревшим NAT. В это время нет никакой поддержки в Cisco IOS для NAT NVI с MPLS.

Развертывания MPLS VPN спроектированы для увеличения быстро за следующие несколько лет. Преимущества инфраструктуры общей сети, которая разрешает быстрое расширение и гибкие параметры подключения, будут, несомненно, вести дальнейший рост в услугах, которые могут быть предложены Межсетевому сообществу.

Однако барьеры к росту все еще остаются. IPv6 и его обещание пространства IP-адресов, которое превышает потребности подключения в обозримом будущем, находятся все еще в ранних фазах развертываний. Существующие сети обычно используют частные IP схемы адресации, как определено в RFC 1918. Когда наложение адресных пространств или дублирование существуют, трансляция сетевых адресов часто используется для соединения сетей.

Поставщики услуг и предприятия, которые имеют сетевые прикладные услуги, которые они хотят предложить или совместно использовать с клиентами и партнерами, захотят минимизировать любую нагрузку подключения, размещенную в пользователя сервиса. Это

является выбираемым, даже обязательным, для расширения предложения как много возможных пользователей по мере необходимости, чтобы достигнуть желаемых целей или возвратиться. Схема IP-адресации в использовании не должна быть барьером, который исключает возможных пользователей.

Путем развертывания ПРЕОБРАЗОВАНИЯ СЕТЕВЫХ АДРЕСОВ В CISCO IOS в общей инфраструктуре MPLS VPN поставщики услуг связи могут уменьшить часть нагрузки подключения на клиентах и ускорить их способность связать сервисы более общего приложения с большим количеством потребителей тех сервисов.

Преимущества от NAT – интеграция MPLS

Интеграция NAT с MPLS обладает преимуществами для обоих поставщиков услуг и их корпоративных клиентов. Это предлагает поставщикам услуг больше опций, чтобы развернуть разделенные сервисы и предоставить доступ к тем сервисам. Предложения дополнительного сервиса могут быть дифференциатором по конкурентам.

Для поставщика услуг	Для VPN
Больше предложений по обслуживанию	Уменьшенные затраты
Увеличенные опции доступа	Более простой доступ
Увеличенный доход	Адресация к гибкости

Корпоративные клиенты, стремящиеся производить часть на стороне их текущей рабочей нагрузки, могут также извлечь выгоду из более широких предложений поставщиками услуг. Смещение нагрузки выполнения любой необходимой переадресации к сети поставщика услуг уменьшает их от сложной административной задачи. Клиенты могут продолжить использовать частную адресацию, все же поддерживать доступ к совместно используемым сервисам и Интернету. Консолидация функции NAT в сети поставщика услуг может также понизить общие затраты для корпоративных клиентов, так как граничные маршрутизаторы клиента не должны выполнять функцию NAT.

Принципы проектирования

При рассмотрении дизайнов, которые вызовут NAT в сети MPLS, первый шаг должен определить сервисные потребности с точки зрения приложения. Необходимо будет считать протоколы используемыми и любая специальная клиент-серверная связь наложенный приложением. Удостоверьтесь, что необходимая поддержка используемых протоколов поддерживается и обрабатываемый ПРЕОБРАЗОВАНИЕМ СЕТЕВЫХ АДРЕСОВ В CISCO IOS. Список поддерживаемых протоколов предоставлен в Шлюзах уровня приложения ПРЕОБРАЗОВАНИЯ СЕТЕВЫХ АДРЕСОВ В CISCO IOS документа.

Затем, будет необходимо определить ожидаемое использование совместно используемого сервиса и ожидаемой скорости трафика в пакетах в секунду. NAT является маршрутизатором, с высокой загрузкой ЦПУ функционируют. Поэтому требования к производительности будут фактором в выборе определенного параметра развертывания и определить количество включенных устройств NAT.

Кроме того, рассмотрите любые проблемы безопасности и меры предосторожности, которые должны быть приняты. Несмотря на то, что MPLS VPN, по определению, являются

частными и эффективно разделяют трафик, совместно используемая сеть услуг обычно распространена среди многих VPN.

Сценарии развертывания

Существует две опции для развертываний NAT в границе провайдера MPLS:

- Централизованный с выходными PE NAT
- Распределенный с входными PE NAT

Некоторые преимущества для настройки функции NAT в исходящей точке сети MPLS, самой близкой к совместно используемой сети услуг, включают:

- Централизованная конфигурация, которая способствует более простой сервисной инициализации
- Упрощенное устранение проблем
- Расширенная в рабочем состоянии масштабируемость
- Уменьшенные требования IP - адреса размещения

Однако преимущества смещены сокращением масштабируемости и производительности. Это - основной компромисс, который нужно рассмотреть. Конечно, функция NAT может также быть выполнена в сетях заказчика, если определено, что интеграция этой функции с сетью MPLS не выбираема.

Входной PE NAT

NAT может быть настроен во входном Периферийном маршрутизаторе сети MPLS как показано на рисунке 1. В то время как производительность оптимизирована путем распределения функции NAT по многим периферийным устройствам, с этим дизайном масштабируемость поддержана в большой степени. Каждый PE NAT обрабатывает трафик для узлов, локально связанных с тем PE. Правила NAT и контроль за списками контроля доступа или Картами маршрутизации, какие пакеты требуют трансляции.

Рисунок 1: Входной PE NAT

Существует ограничение, которое предотвращает NAT между двумя VRF, также предоставляя NAT совместно используемому сервису как показано на рисунке 2. Это происходит из-за требования для обозначения интерфейсов как NAT “внутри” и “вне” интерфейсов. Поддержка соединений между VRF в одиночном PE запланирована будущий Cisco IOS Release.

Рис. 2: Предоставление товаров и услуг предприятиям

Выходной PE NAT

NAT может быть настроен в выходном Периферийном маршрутизаторе сети MPLS как показано на рисунке 3. С этим дизайном масштабируемость уменьшена до некоторой степени, так как центральный PE должен поддержать маршруты для всех сетей заказчика, которые обращаются к совместно используемому сервису. Требования приложений к производительности нужно также рассмотреть так, чтобы трафик не перегружал маршрутизатор, который должен преобразовать IP-адреса пакетов. Поскольку NAT происходит централизованно для всех клиентов, использующих этот путь, пулы IP-адреса могут быть разделены; таким образом общее число требуемых подсетей уменьшено.

Рис. 3: Выходной PE NAT

Несколька маршрутизаторов могли быть развернуты для увеличения масштабируемости выходного PE дизайн NAT как показано на рисунке 4. В этом сценарии VPN клиента могли быть “настроены” на определенном маршрутизаторе NAT. Трансляция сетевых адресов произошла бы для совокупного трафика с и от совместно используемого сервиса для того набора VPN. Например, в то время как трафик к и от VPN для клиента К использует PE2 NAT, трафик от VPN для Клиента А и В мог использовать PE1 NAT. Каждый PE NAT нес бы трафик только для определенных определенных VPN и только поддержал бы маршруты назад к узлам в тех VPN. Отдельные Пулы адресов NAT могли быть определены в каждом из Периферийных маршрутизаторов NAT так, чтобы пакеты маршрутизировались от совместно используемой сети услуг до надлежащего PE NAT для трансляции и направляющий назад к VPN клиента.

Рис. 4: Множественный выходной PE NAT

Централизованный дизайн действительно вводит ограничение для того, как должна быть настроена совместно используемая сеть услуг. В частности использование импорта/экспорта маршрутов MPLS VPN между совместно используемой сервисной VPN и VPN клиента не возможно. Это происходит из-за природы операции MPLS, как задано RFC 2547. Когда маршруты импортированы и экспортировали использование расширенных сообществ и дескрипторов маршрута, NAT не может определить исходную VPN от пакета, войдя в центральный PE NAT. Обычный случай должен сделать совместно используемую сеть услуг общим интерфейсом, а не интерфейсом VRF. Маршрут к совместно используемой сети услуг тогда добавлен в центральном PE NAT для каждой таблицы VRF, привязанной к доступу необходимости VPN клиента к совместно используемому сервису как часть процесса инициализации. Это описано более подробно позже.

Параметры развертывания и элементы конфигурации

Этот раздел включает некоторые подробные данные, отнесенные в каждый из параметров развертывания. Примеры все взяты от сети, показанной на рисунке 5. См. эту схему для остатка этого раздела.

Примечание: В сети, используемой для иллюстрирования использования VRF NAT для этой бумаги только включены Периферийные маршрутизаторы. Нет никакого ядра “P” маршрутизаторов. Однако существенные механизмы могут все еще быть замечены.

Рис. 5: Пример конфигурации NAT VRF

Выходной PE NAT

В данном примере маршрутизаторы на стороне провайдера отметили **Хилу**, и **дракон** настроены как простые Периферийные маршрутизаторы. Центральный PE около совместно используемой сервисной LAN **(игуана)** настроен для NAT. Одиночный пул NAT разделен каждой VPN клиента, которая должна обратиться к совместно используемому сервису. NAT выполнен только на пакетах, предназначенных для совместно используемого сервисного хоста в 88.1.88.8.

Выходной PE переадресация данных NAT

С MPLS каждый пакет вводит сеть во входной PE и выходит из сети MPLS в выходном PE. Путь Маршрутизаторов коммутации меток, пересеченных от входа до выхода, известен как

путь коммутации меток (LSP). LSP однонаправлен. Другой LSP используется для ответного трафика.

При использовании выходного PE NAT Forwarding Equivalence Class (FEC) эффективно определен для всего трафика от пользователей совместно используемого сервиса. Другими словами, все пакеты, предназначенные для совместно используемой сервисной LAN, являются участниками общего FEC. Пакет назначен на определенный FEC только однажды в границе на входе сети и придерживается LSP к выходному PE. FEC определяется в пакете данных путем добавления определенной метки.

Поток пакетов к совместно используемому сервису от VPN

Для устройств во множественных VPN, которые имеют схемы совмещенного адреса обратиться к совместно используемому сервисному хосту, требуется NAT. Когда NAT будет настроен в выходном PE, элементы таблицы трансляции сетевых адресов будут включать идентификатор VRF, чтобы дифференцировать дублирования адреса и гарантировать соответствующую маршрутизацию.

Рис. 6: Пакеты, переданные к выходному PE NAT

Рисунок 6 иллюстрирует пакеты, предназначенные для совместно используемого сервисного хоста от двух VPN клиента, которые имеют схемы адресации IP - адресации с дублированием. Данные показывают пакет, происходящий в Клиенте с адресом источника 172.31.1.1 предназначенных для совместно используемого сервера в 88.1.88.8. Другой пакет от Клиента Б с тем же IP - адресом источника также передан к тому же совместно используемому серверу. Когда пакеты достигают Периферийного маршрутизатора, поиск уровня 3 сделан для сети IP - адреса назначения в базе данных переадресации (FIB).

Запись FIB говорит Периферийному маршрутизатору передавать трафик к выходному PE с помощью стека меток. Нижняя метка в стеке назначена целевым Периферийным маршрутизатором в этой **игуане** маршрутизатора случая.

```
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#
```

Мы видим от показа, что пакеты от VRF custA будут иметь значение метки 24 (0x18), и пакеты от VRF custB будут иметь значение метки 28 (0x1C).

В этом случае, потому что нет никаких маршрутизаторов “P” в нашей сети, нет никакой дополнительной наложенной метки. Были центральные маршрутизаторы, внешняя метка будет наложена, и обычный процесс свопинга метки имел бы место в базовой сети, пока пакет не достиг выходного PE.

Так как маршрутизатор **Хилы** напрямую подключается к выходному PE, мы видим, что метка вытолкана, прежде чем это будет когда-либо добавлено:

```
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#

Следующие показы изображают эхо - пакеты, как получено выходным маршрутизатором NAT PE (в интерфейсном E1/0/5 на **игуане**).

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".]

Эти эхо-запросы приводят к следующим записям, создаваемым в таблице NAT в выходной **игуане** Периферийного маршрутизатора. Со специальными записями, созданными для пакетов, показанных выше, может совпасть их идентификатор 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#
```

Поток пакетов от совместно используемого сервиса Назад к VPN происхождения

Поскольку пакеты текут назад к устройствам, которые обратились к совместно используемому сервисному хосту, таблица NAT исследована до маршрутизации (пакеты, идущие от NAT “вне” интерфейса к “внутреннему” интерфейсу). Поскольку каждая уникальная запись включает соответствующий идентификатор VRF, пакет может преобразовываться и маршрутизироваться соответственно.

Рисунок 7: Пакеты, переданные Назад совместно используемому пользователю службы

Как показано на рисунке 7, ответный трафик сначала исследован NAT для обнаружения соответствующего транслируемого значения. Например, пакет передан назначению 192.168.1.1. Таблица NAT ищется. Когда соответствие найдено, соответствующая запись о трансляции сделана к “в локальном” адресе (172.31.1.1), и затем поиск смежности выполнен с помощью связанного ID VRF от Записи 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#
```

Метка 23 (0x17) используется для трафика, предназначенного для 172.31.1.0/24 в VRF custA, и метка 26 (0x1A) используется для пакетов, предназначенных для 172.31.1.0/24 в VRF custB.

Это замечено в пакетах эхо-ответа, передаваемых от **игуаны** маршрутизатора:

```
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".]
```

Когда пакет достигает целевого Периферийного маршрутизатора, метка используется для

определения соответствующего VRF и интерфейса для передачи пакета.

```
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#
```

Конфигурации

Некоторая посторонняя информация была удалена из конфигураций для краткости.

```
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
!
```

У дракона маршрутизатора была бы конфигурация подобной **Хиле**.

Импорт/Экспорт целей Маршрута, Не Разрешенных

Когда совместно используемая сеть услуг настроена как сам экземпляр VRF, центральный NAT в выходном PE не возможен. Это вызвано тем, что входящие пакеты нельзя отличить, и только один маршрут назад к инициирующей подсети присутствует в выходном PE NAT.

Примечание: Показы, которые придерживаются, предназначаются для иллюстрирования результата неправильной конфигурации.

Пример сети был настроен так, чтобы совместно используемая сеть услуг была определена как экземпляр VRF (Имя VRF = сервер). Теперь, показ таблицы CEF на входном PE показывает это:

```
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#
```

Примечание: Заметьте, как значение метки *24* наложено и для VRF custA и для VRF custB.

Этот показ показывает таблицу маршрутизации для совместно используемого сервисного экземпляра VRF “сервер”:

```
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
```

Примечание: Только один маршрут присутствует для сети назначения от выходного Периферийного маршрутизатора **(игуана)** перспектива.

Поэтому трафик от VPN несколька пользовательских устройств нельзя было отличить, и ответный трафик не мог достигнуть соответствующей VPN. **В случае, где совместно используемый сервис должен быть определен как экземпляр VRF, функция NAT должна быть перемещена во входной PE.**

Входной PE NAT

В данном примере маршрутизаторы на стороне провайдера отметили **Хилу**, и **дракон** настроены для NAT. Пул NAT определен для каждой подключенной VPN клиента, которая должна обратиться к совместно используемому сервису. Соответствующий пул используется для каждого из адресов сети заказчика, которые преобразованы посредством NAT. NAT выполнен только на пакетах, предназначенных для совместно используемого сервисного хоста в 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
```

Примечание: В этом сценарии не поддерживаются разделенные пулы. Если совместно используемая сервисная LAN (в выходном PE) связана через общий интерфейс, то пул NAT может быть разделен.

Эхо-запрос, полученный от дублирования адреса (172.31.1.1) в каждой из сетей, подключил к **neuse** и результатам **capefear8** в этих Записях NAT:

Из **Хилы**:

```
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
```

Примечание: Тот же внутренний локальный адрес (172.31.1.1) преобразован в каждый из определенных пулов согласно исходному VRF. VRF может быть замечен в **команде show ip nat translation verbose**:

```
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
```

Эти показы показывают сведения о маршрутизации для каждой из локально подключенных VPN для клиента А и клиента Б:

```
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
```

Примечание: Маршрут для каждого из пулов NAT был добавлен от статичной конфигурации. Эти подсети впоследствии импортированы в совместно используемый VRF сервера в выходной **игуане** Периферийного маршрутизатора:

```
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
```

Конфигурации

Некоторая посторонняя информация была удалена из конфигураций для краткости.

```
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 !
```

Примечание: Интерфейсы, которые стоят перед сетями заказчика, определяются как NAT “в” интерфейсах, и интерфейсы MPLS определяются как NAT “вне” интерфейсов.

```
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
```

У дракона маршрутизатора была бы конфигурация подобной **Хиле**.

Пакеты, Поступающие в Центральный PE после Входного PE NAT

Когда разделенная сеть услуг назначения настроена как экземпляр VRF, трассировки ниже иллюстрируют требование для уникальных пулов NAT. Снова, обратитесь к схеме на рисунке 5. Пакеты, показанные ниже, были перехвачены, когда они ввели IP - интерфейс MPLS e1/0/5 в **игуану** маршрутизатора.

Эхо от клиента VPN

Здесь, мы видим, что запрос эха прибывает из IP - адреса источника 172.31.1.1 в VRF custA. Адрес источника был преобразован в 192.168.1.1, как задано конфигурацией 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:
```

Эхо от VPN клиента Б

Здесь, мы видим, что запрос эха прибывает из IP - адреса источника 172.31.1.1 в VRF custB. Адрес источника был преобразован в 192.168.2.1, как задано конфигурацией 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".]
```

Примечание: Значение MPLS label *0019* в обоих из пакетов, показанных выше.

Эхо - ответ клиенту VPN

Затем, мы видим, что эхо - ответ возвращается к IP - адресу назначения 192.168.1.1 в VRF custA. Адрес назначения (DA) преобразован в 172.31.1.1 входной функцией 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:
```

Эхо - ответ к VPN клиента Б

Здесь, мы видим, что эхо - ответ возвращается к IP - адресу назначения 192.168.1.1 в VRF custB. Адрес назначения (DA) преобразован в 172.31.1.1 входной функцией 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".]
```

Примечание: В возвращаемых пакетах значения MPLS label включены и отличаются: *001A* для VRF custA и *001D* для VRF custB.

Эхом от клиента VPN – назначение является общий интерфейс

Когда интерфейс к совместно используемой сервисной LAN является общим интерфейсом и не частью экземпляра VRF, этот следующий набор пакетов показывает различие. Здесь,

конфигурация была изменена для использования общего пула для обеих локальных VPN с перекрывающимися IP-адресами.

```
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".]
```

Эхом от VPN клиента Б – назначение является общий интерфейс

Здесь, мы видим, что запрос эха прибывает из IP - адреса источника 172.31.1.1 в VRF custB. Адрес источника был преобразован в 192.168.1.3 (от общего пула SSPOOL1), как задано конфигурацией 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".]
```

Примечание: Когда интерфейс в выходном PE является общим интерфейсом (не экземпляр VRF), наложенные метки являются другими. В этом случае, *0x19* и *0x1F*.

Эхом - ответом клиенту VPN – назначение является общий интерфейс

Затем, мы видим, что эхо - ответ возвращается к IP - адресу назначения 192.168.1.1 в VRF custA. Адрес назначения (DA) преобразован в 172.31.1.1 входной функцией 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".]
```

Эхом - ответом к VPN клиента Б – назначение является общий интерфейс

Здесь, мы видим, что эхо - ответ возвращается к IP - адресу назначения 192.168.1.3 в VRF custB. Адрес назначения (DA) преобразован в 172.31.1.1 входной функцией 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".]
```

Примечание: Так как ответы предназначены к глобальному адресу, никакие метки VRF не наложены.

С выходным интерфейсом к совместно используемому сервисному сегменту LAN, определенному как общий интерфейс, разрешен общий пул. Эхо-запросы приводят к этим Записям NAT в маршрутизаторе **Хила**:

```
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#
```

Сервисный пример

Пример совместно используемого виртуального IP сервиса УАТС показывают на рисунке 8. Это иллюстрирует вариант к входным и выходным примерам, описанным ранее.

В этом дизайне совместно используемый Сервис VoIP закончен передней стороной рядом маршрутизаторов, которые выполняют функцию NAT. Эти маршрутизаторы имеют множественные интерфейсы VRF, использующие функцию, известную как Облегченные VRF. Трафик тогда течет к совместно используемому Кластеру Cisco CallManager. Сервисы межсетевого экрана также предоставлены на основе на компанию. В то время как вызовы внутрикомпании обрабатываются через VPN клиента с помощью внутренней схемы адресации компании, межфирменные вызовы должны пройти через межсетевой экран.

Рис. 8: Управляемый действительный пример сервиса УАТС

Доступность

Поддержка ПРЕОБРАЗОВАНИЯ СЕТЕВЫХ АДРЕСОВ В CISCO IOS MPLS VPN доступна в Cisco IOS Release 12.2 (13) T и доступна для всех платформ, которые поддерживают MPLS и могут выполнить эту серию релиза раннего развертывания.

Заключение

ПРЕОБРАЗОВАНИЕ СЕТЕВЫХ АДРЕСОВ В CISCO IOS имеет функции для разрешения масштабируемого развертывания совместно используемых сервисов сегодня. Cisco продолжает разрабатывать поддержку шлюза прикладного уровня (ALG) NAT протоколов, важных для клиентов. Повышения производительности и аппаратное ускорение для функций преобразования гарантируют, что NAT и ALGs предоставляют приемлемые решения в течение некоторого времени. Все действия соответствующих стандартов и общественные действия проверяются Cisco. Поскольку другие стандарты разработаны, их использование будет оценено на основе желаний клиента, требований и приложения.

Дополнительные сведения

- Шлюзы уровня приложения ПРЕОБРАЗОВАНИЯ СЕТЕВЫХ АДРЕСОВ В CISCO IOS
- MPLS и архитектуры VPN
- Усовершенствованная разработка и реализация MPLS
- Cisco Systems – техническая поддержка и документация