Cisco IOS NAT - Integrazione con MPLS VPN

Sommario

Introduzione

Il software Cisco IOS® Network Address Translation (NAT) consente di accedere ai servizi condivisi da più VPN MPLS, anche quando i dispositivi delle VPN utilizzano indirizzi IP che si sovrappongono. Cisco IOS NAT è compatibile con VRF e può essere configurato sui router periferici del provider all'interno della rete MPLS.

Nota: MPLS in IOS è supportato solo con NAT legacy. Al momento, Cisco IOS non supporta NAT NVI con MPLS.

L'implementazione delle VPN MPLS dovrebbe aumentare rapidamente nei prossimi anni. I vantaggi di un'infrastruttura di rete comune che consenta una rapida espansione e opzioni di connettività flessibili contribuiranno senza dubbio a un'ulteriore crescita dei servizi che possono essere offerti alla comunità di internetworking.

Restano tuttavia ostacoli alla crescita. L'IPv6 e la promessa di uno spazio di indirizzi IP che superi le esigenze di connettività per il prossimo futuro sono ancora nelle prime fasi dell'installazione. Le reti esistenti in genere utilizzano schemi di indirizzamento IP privato definiti nella RFC 1918 . La conversione degli indirizzi di rete viene spesso utilizzata per interconnettere le reti quando gli spazi degli indirizzi si sovrappongono o sono presenti duplicazioni.

I fornitori di servizi e le aziende che dispongono di servizi di applicazioni di rete che desiderano offrire o condividere con clienti e partner desiderano ridurre al minimo l'onere di connettività per l'utente del servizio. È auspicabile, se non obbligatorio, estendere l'offerta a tutti i potenziali utenti necessari per raggiungere gli obiettivi desiderati o per ottenere un ritorno. Lo schema di indirizzamento IP in uso non deve costituire una barriera che escluda gli utenti potenziali.

Implementando Cisco IOS NAT nell'infrastruttura VPN MPLS comune, i provider di servizi di

comunicazione possono ridurre parte del carico di connettività per i clienti e accelerare la loro capacità di collegare più servizi applicativi condivisi a più utenti di tali servizi.

Vantaggi dell'integrazione NAT - MPLS

L'integrazione NAT con MPLS offre vantaggi sia ai provider di servizi che ai clienti aziendali. Offre ai provider di servizi più opzioni per distribuire servizi condivisi e fornire accesso a tali servizi. Le offerte di servizi aggiuntivi possono essere un fattore di differenziazione rispetto alla concorrenza.

Per provider di servizi	Per VPN
Altre offerte di servizi	Riduzione dei costi
Maggiori opzioni di accesso	Accesso più semplice
Incremento dei profitti	Flessibilità

Anche i clienti aziendali che desiderano esternalizzare parte del carico di lavoro corrente possono trarre vantaggio da offerte più ampie da parte dei provider di servizi. Spostare l'onere della traduzione dell'indirizzo necessario alla rete del provider di servizi li solleva da una complicata attività amministrativa. I clienti possono continuare a utilizzare gli indirizzi privati, pur mantenendo l'accesso ai servizi condivisi e a Internet. Il consolidamento della funzione NAT all'interno della rete del provider di servizi può anche ridurre il costo totale per i clienti aziendali, in quanto i router periferici del cliente non devono eseguire la funzione NAT.

Considerazioni sulla progettazione

Quando si prendono in considerazione progetti che richiameranno NAT all'interno della rete MPLS, il primo passo è determinare le esigenze di servizio da un punto di vista dell'applicazione. È necessario prendere in considerazione i protocolli utilizzati e qualsiasi comunicazione client/server speciale imposta dall'applicazione. Accertarsi che il supporto necessario per i protocolli utilizzati sia supportato e gestito da Cisco IOS NAT. Nel documento Cisco IOS NAT Application Layer Gateway è fornito un elenco dei protocolli supportati.

Sarà quindi necessario determinare l'utilizzo previsto del servizio condiviso e la velocità prevista del traffico in pacchetti al secondo. NAT è una funzione che richiede un uso intensivo della CPU del router. Pertanto, i requisiti di prestazioni saranno un fattore nella selezione di una particolare opzione di distribuzione e per determinare il numero di dispositivi NAT coinvolti.

Inoltre, considera eventuali problemi di sicurezza e precauzioni da adottare. Sebbene le VPN MPLS, per definizione, siano private e separino in modo efficace il traffico, la rete di servizio condiviso è generalmente comune tra molte VPN.

Scenari di distribuzione

Per l'implementazione NAT all'interno del perimetro del provider MPLS sono disponibili due opzioni:

- Centralizzato con PE NAT in uscita
- Distribuito con PE NAT in entrata

Alcuni vantaggi della configurazione della funzione NAT sul punto di uscita della rete MPLS più vicina alla rete di servizi condivisi sono:

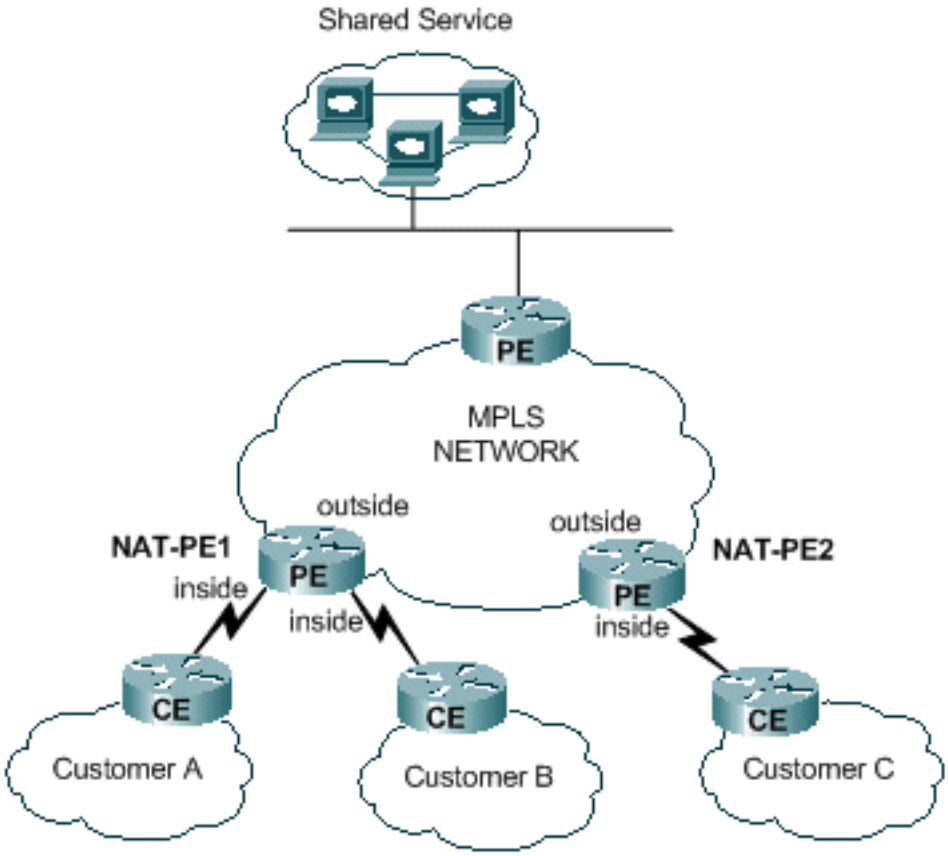
- Una configurazione centralizzata che semplifica il provisioning dei servizi
- Risoluzione dei problemi semplificata
- Maggiore scalabilità operativa
- Riduzione dei requisiti di allocazione degli indirizzi IP

Tuttavia, i vantaggi sono compensati da una riduzione della scalabilità e delle prestazioni. Questo è il principale compromesso da prendere in considerazione. Naturalmente, la funzione NAT può essere eseguita anche all'interno delle reti del cliente se non è consigliabile integrare questa funzione con una rete MPLS.

NAT PE in ingresso

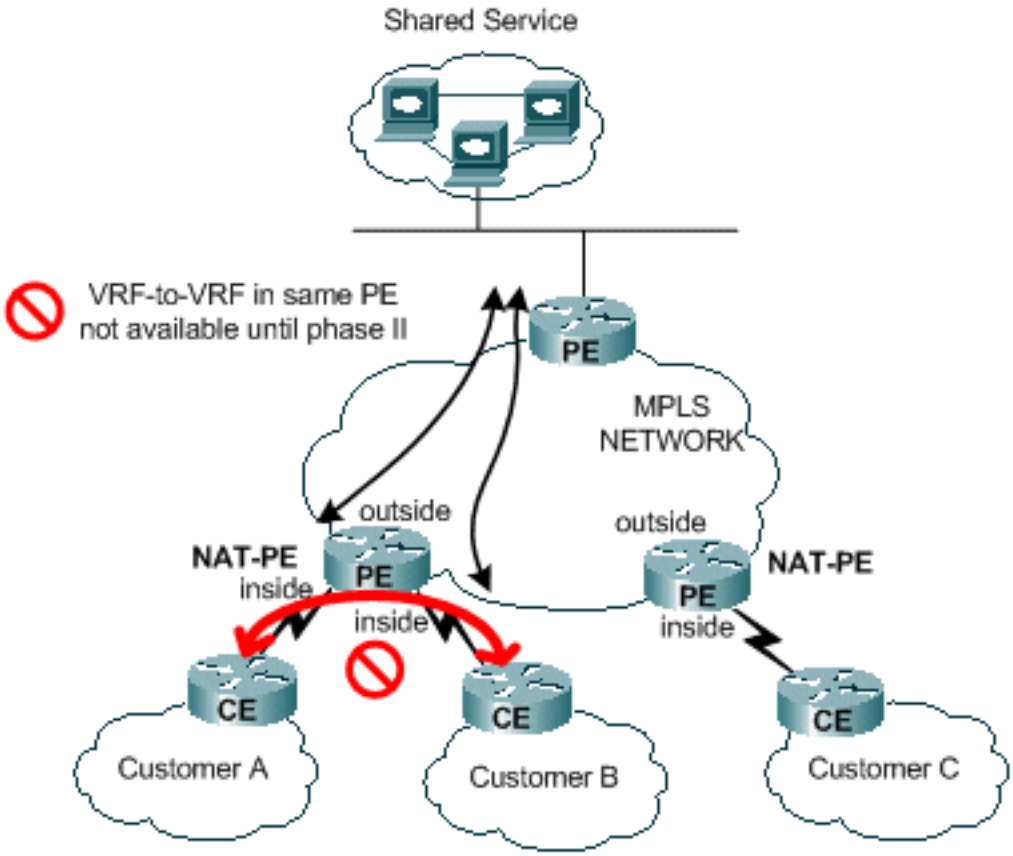
Èpossibile configurare NAT sul router PE in entrata nella rete MPLS, come mostrato nella Figura 1. Con questo progetto, la scalabilità viene mantenuta in larga misura, mentre le prestazioni vengono ottimizzate distribuendo la funzione NAT su molti dispositivi periferici. Ogni NAT PE gestisce il traffico per i siti connessi localmente a tale PE. Le regole NAT e gli elenchi di controllo degli accessi o le mappe dei percorsi controllano quali pacchetti devono essere tradotti.

Figura 1: NAT PE in ingresso



Esiste una restrizione che impedisce il NAT tra due VRF e nel contempo fornisce NAT a un servizio condiviso, come mostrato nella Figura 2. Ciò è dovuto alla necessità di designare le interfacce come interfacce NAT "interne" ed "esterne". Il supporto per le connessioni tra VRF in un singolo PE è pianificato per una futura versione di Cisco IOS.

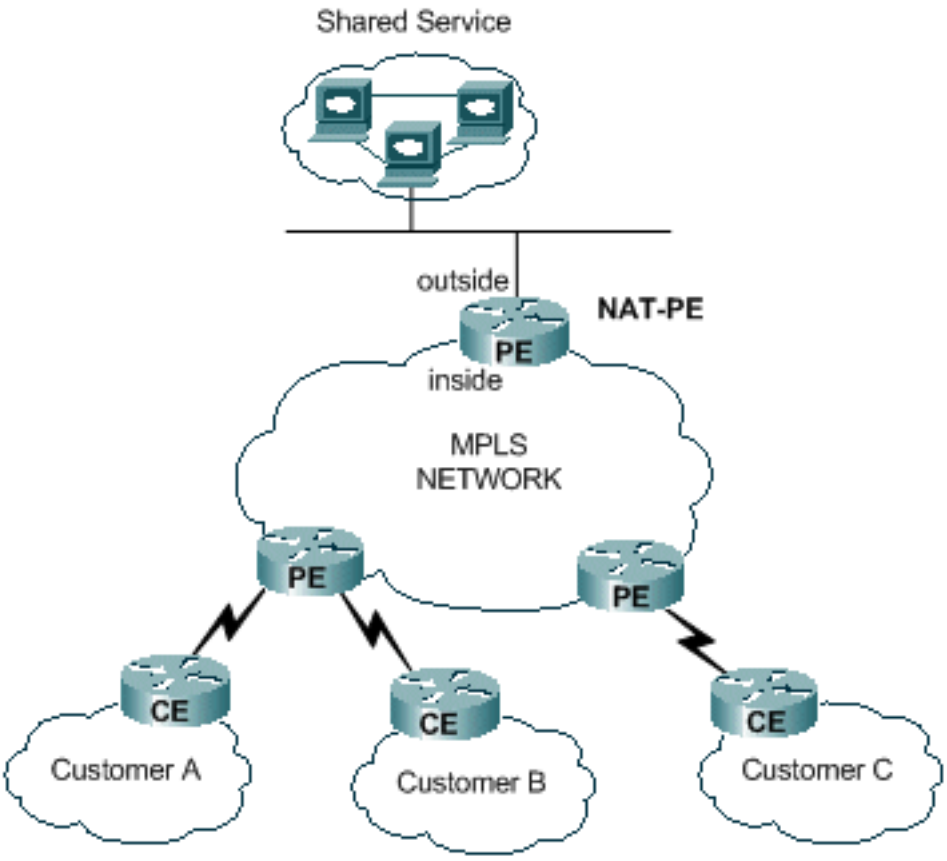
Figura 2: Business-to-Business



Uscita PE NAT

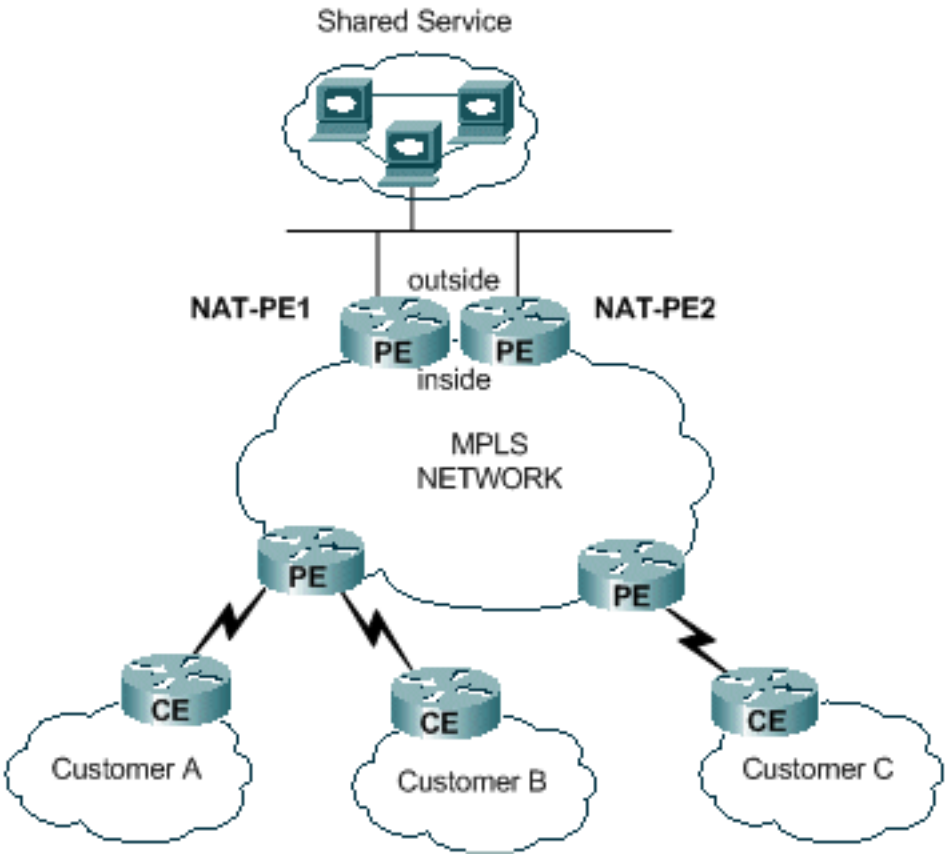
Èpossibile configurare NAT sul router PE in uscita della rete MPLS, come mostrato nella Figura 3. Con questa progettazione, la scalabilità è ridotta in qualche misura in quanto il PE centrale deve mantenere i percorsi per tutte le reti del cliente che accedono al servizio condiviso. Inoltre, è necessario tenere in considerazione i requisiti di prestazioni delle applicazioni, in modo che il traffico non sovraccarichi il router che deve convertire gli indirizzi IP dei pacchetti. Poiché NAT si verifica a livello centrale per tutti i clienti che utilizzano questo percorso, è possibile condividere i pool di indirizzi IP. il numero totale di subnet richieste risulta pertanto ridotto.

Figura 3: Uscita PE NAT



Èpossibile implementare più router per aumentare la scalabilità del progetto NAT PE in uscita, come mostrato nella Figura 4. In questo scenario, è possibile eseguire il "provisioning" delle VPN del cliente su un router NAT specifico. La conversione degli indirizzi di rete verrebbe eseguita per il traffico aggregato da e verso il servizio condiviso per il set di VPN specificato. Ad esempio, il traffico proveniente dalle VPN per i clienti A e B potrebbe utilizzare NAT-PE1, mentre il traffico diretto alla VPN e proveniente dalla VPN per i clienti C potrebbe utilizzare NAT-PE2. Ogni NAT-PE trasporterebbe il traffico solo per le VPN specifiche definite e gestirebbe solo le route verso i siti di tali VPN. È possibile definire pool di indirizzi NAT distinti all'interno di ogni router NAT PE in modo che i pacchetti vengano instradati dalla rete di servizi condivisi al server NAT PE appropriato per la conversione e il routing alla VPN del cliente.

Figura 4: NAT PE in uscita multipla



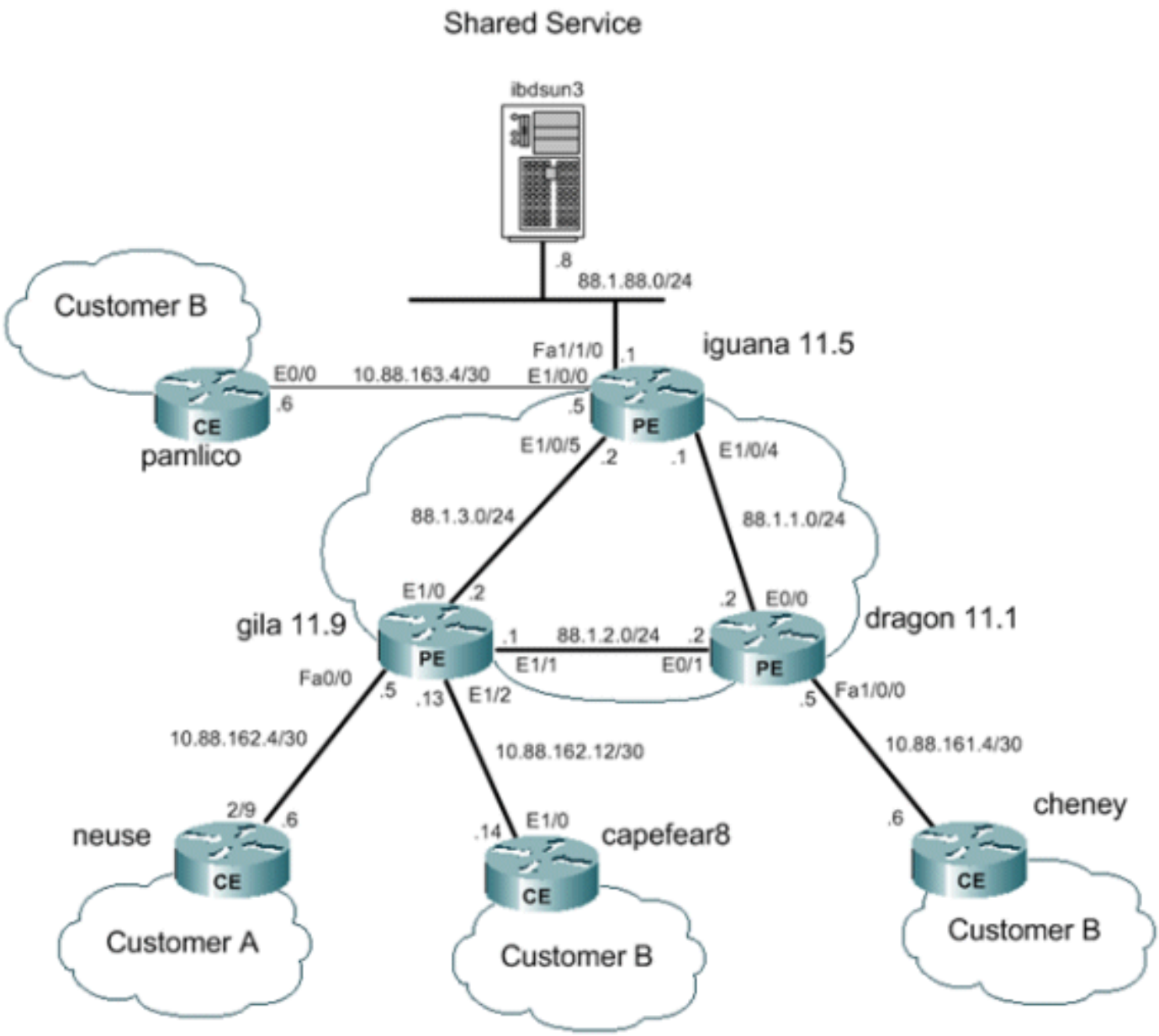
La progettazione centralizzata impone una restrizione sulla modalità di configurazione della rete di servizi condivisi. In particolare, non è possibile utilizzare l'importazione/esportazione di route VPN MPLS tra una VPN a servizio condiviso e le VPN del cliente. Ciò è dovuto alla natura del funzionamento di MPLS come specificato nella RFC 2547 . Quando le route vengono importate ed esportate utilizzando le community estese e i descrittori di route, NAT non è in grado di determinare la VPN di origine dal pacchetto in arrivo nel PE NAT centrale. In genere, la rete di servizi condivisi viene utilizzata come interfaccia generica anziché come interfaccia VRF. Viene quindi aggiunto un percorso alla rete di servizi condivisi nella tabella centrale NAT PE per ogni tabella VRF associata a una VPN del cliente che richiede l'accesso al servizio condiviso come parte del processo di provisioning. Questo viene descritto più dettagliatamente in seguito.

Opzioni di distribuzione e dettagli di configurazione

In questa sezione sono inclusi alcuni dettagli correlati a ciascuna opzione di distribuzione. Gli esempi sono tutti tratti dalla rete mostrata nella Figura 5. Per il resto, fare riferimento a questo diagramma.

Nota: nella rete utilizzata per illustrare il funzionamento del VRF NAT per questo documento, sono inclusi solo i router PE. Non sono presenti router "P" di base. Tuttavia, i meccanismi essenziali sono ancora visibili.

Figura 5: Esempio di configurazione VRF NAT



Uscita PE NAT

In questo esempio, i router periferici del provider contrassegnati come **gila** e **dragon** sono configurati come router PE semplici. Il sistema PE centrale vicino alla LAN del servizio condiviso (**iguana**) è configurato per NAT. Un singolo pool NAT viene condiviso da ciascuna VPN del cliente che deve accedere al servizio condiviso. Il protocollo NAT viene eseguito solo sui pacchetti destinati all'host del servizio condiviso in modalità 88.1.88.8.

Inoltro dati PE NAT in uscita

Con MPLS, ogni pacchetto entra nella rete in ingresso e esce dalla rete MPLS in uscita. Il percorso dei router di commutazione di etichetta attraversati da in entrata a in uscita è noto come percorso a commutazione di etichetta (LSP). L'LSP è unidirezionale. Per il traffico di ritorno viene utilizzato un altro provider di servizi di traduzione.

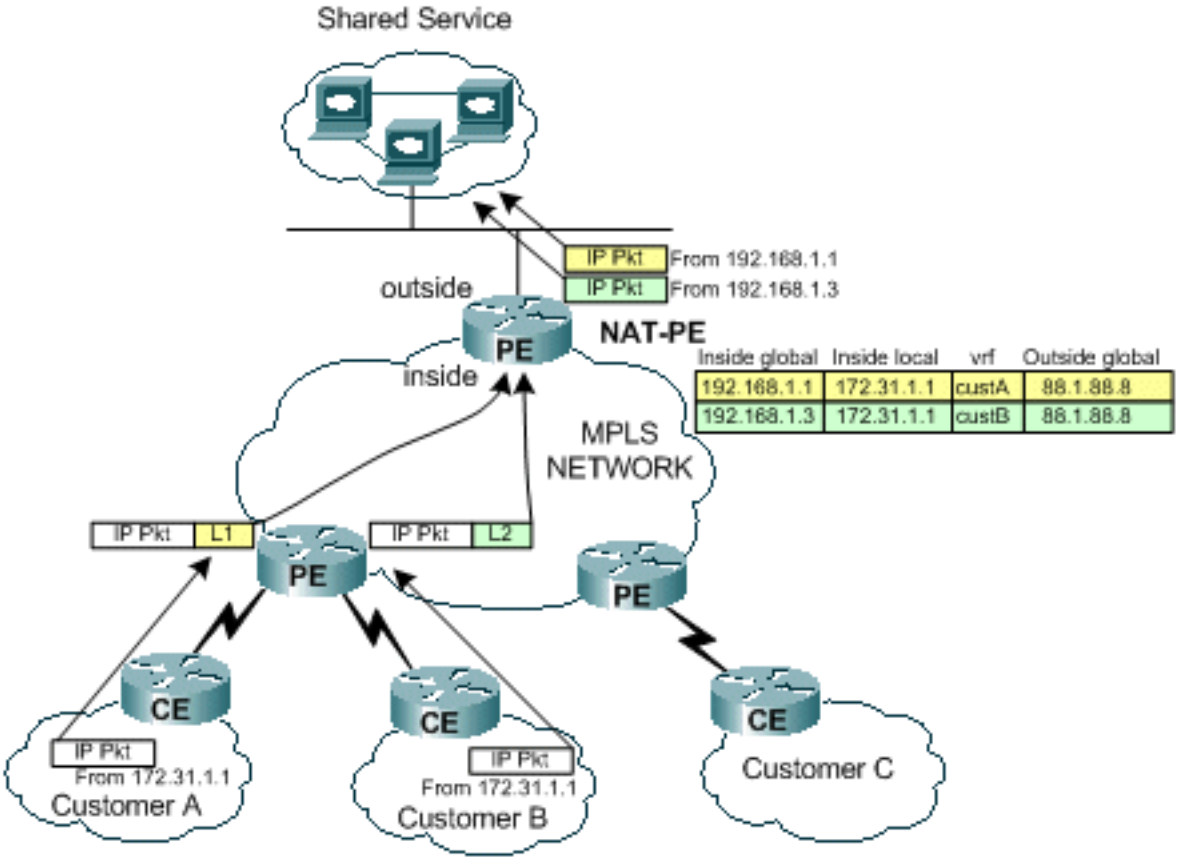
Quando si utilizza il protocollo PE NAT in uscita, viene effettivamente definita una classe di equivalenza dell'inoltro (FEC) per tutto il traffico proveniente dagli utenti del servizio condiviso. In altre parole, tutti i pacchetti destinati alla LAN a servizio condiviso sono membri di una FEC comune. Un pacchetto viene assegnato a una determinata unità FEC una sola volta in

corrispondenza del margine di ingresso della rete e segue l'LSP fino al punto di uscita. La funzione FEC viene indicata nel pacchetto aggiungendo un'etichetta specifica.

Flusso di pacchetti al servizio condiviso dalla VPN

Per consentire ai dispositivi di più VPN con schemi di indirizzi sovrapposti di accedere a un host del servizio condiviso, è necessario NAT. Quando NAT è configurato in corrispondenza del PE di uscita, le voci della tabella di conversione degli indirizzi di rete includeranno un identificatore VRF per distinguere gli indirizzi duplicati e garantire il routing corretto.

Figura 6: Pacchetti trasmessi a PE NAT in uscita



Nella Figura 6 vengono illustrati i pacchetti destinati a un host di servizi condivisi da due VPN del cliente con schemi di indirizzamento IP duplicati. Nella figura viene mostrato un pacchetto proveniente dal Cliente A con indirizzo di origine 172.31.1.1 e destinato a un server condiviso alla posizione 88.1.88.8. Allo stesso server condiviso viene inviato anche un altro pacchetto proveniente dal Cliente B con lo stesso indirizzo IP di origine. Quando i pacchetti raggiungono il router PE, viene eseguita una ricerca di livello 3 per la rete IP di destinazione nella base di informazioni di inoltro (FIB).

La voce FIB indica al router PE di inoltrare il traffico al PE di uscita utilizzando uno stack di etichette. L'etichetta inferiore nello stack viene assegnata dal router PE di destinazione, in questo caso **iguana** router.

```
iguana#
show ip cef vrf custA 88.1.88.8
88.1.88.8/32, version 47, epoch 0, cached adjacency 88.1.3.2
0 packets, 0 bytes
  tag information set
    local tag: VPN-route-head
    fast tag rewrite with Et1/0, 88.1.3.2, tags imposed: {24}
```

```
  via 88.1.11.5, 0 dependencies, recursive
    next hop 88.1.3.2, Ethernet1/0 via 88.1.11.5/32
    valid cached adjacency
    tag rewrite with Et1/0, 88.1.3.2, tags imposed: {24}
```

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

Dallo schermo si evince che i pacchetti provenienti dall'account VRF avranno un valore di tag pari a 24 (0x18), mentre i pacchetti provenienti dall'account VRF avranno un valore di tag pari a 28 (0x1C).

In questo caso, poiché nella rete non sono presenti router "P", non viene imposto alcun tag aggiuntivo. Se ci fossero stati router principali, sarebbe stata imposta un'etichetta esterna e il normale processo di scambio delle etichette avrebbe avuto luogo all'interno della rete principale fino a quando il pacchetto non raggiungeva il PE in uscita.

Poiché il router **gila** è collegato direttamente al PE in uscita, si noti che il tag viene inserito prima di essere aggiunto:

```
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#
```

Nelle schermate successive vengono mostrati i pacchetti echo ricevuti dal router PE NAT in uscita (sull'interfaccia E1/0/5 sull'interfaccia 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".]
```

Di conseguenza, nella tabella NAT dell'iguana del router PE in uscita vengono create le voci seguenti. Alle voci specifiche create per i pacchetti mostrati sopra può corrispondere il relativo identificatore ICMP.

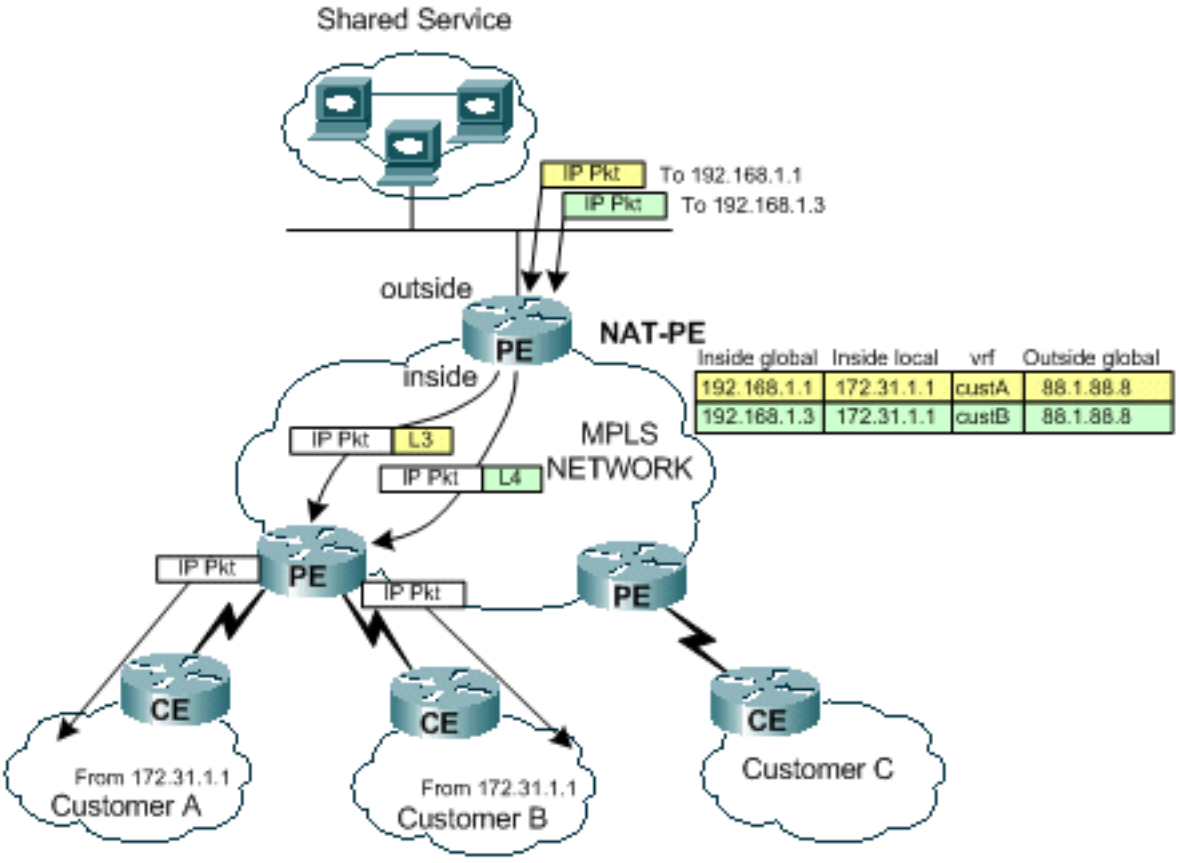
```
iguana#
show ip nat translations
```

```
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#
```

Flusso di pacchetti dal servizio condiviso alla VPN di origine

Quando i pacchetti ritornano ai dispositivi che hanno accesso all'host del servizio condiviso, la tabella NAT viene esaminata prima del routing (i pacchetti vanno dall'interfaccia "esterna" NAT all'interfaccia "interna"). Poiché ogni voce univoca include il corrispondente identificatore VRF, il pacchetto può essere tradotto e indirizzato in modo appropriato.

Figura 7: Pacchetti ritrasmessi all'utente del servizio condiviso



Come mostrato nella Figura 7, il traffico di ritorno viene esaminato da NAT per trovare una voce di traduzione corrispondente. Ad esempio, un pacchetto viene inviato alla destinazione 192.168.1.1. Viene eseguita la ricerca nella tabella NAT. Quando la corrispondenza viene trovata, la traduzione appropriata viene eseguita all'indirizzo "locale interno" (172.31.1.1), quindi viene eseguita una ricerca adiacente utilizzando l'ID VRF associato dalla voce 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#
```

L'etichetta 23 (0x17) viene usata per il traffico destinato a 172.31.1.0/24 nella busta VRF A e l'etichetta 26 (0x1A) viene usata per i pacchetti destinati a 172.31.1.0/24 nella busta VRF B.

Ciò si verifica nei pacchetti di risposta echo inviati dal router **iguana**:

```
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 il pacchetto raggiunge il router PE di destinazione, l'etichetta viene usata per determinare il VRF e l'interfaccia appropriati per inviare il pacchetto.

```
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#
```

Configurazioni

Alcune informazioni estranee sono state rimosse dalle configurazioni per brevità.

```
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
!
```

Il **drago** del router avrebbe una configurazione molto simile a **gila**.

Importazione/esportazione di destinazioni ciclo di lavorazione non consentita

Quando la rete di servizi condivisi è configurata come istanza VRF stessa, non è possibile utilizzare un NAT centrale nel PE di uscita. Questo accade perché i pacchetti in entrata non possono essere distinti e sul NAT del PE in uscita è presente un solo percorso verso la subnet di origine.

Nota: le seguenti visualizzazioni hanno lo scopo di illustrare il risultato di una configurazione non valida.

La rete di esempio è stata configurata in modo che la rete del servizio condiviso sia stata definita come istanza VRF (nome VRF = server). Ora, una visualizzazione della tabella CEF sul PE in entrata mostra questo:

```
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: il valore di tag *24* viene imposto sia per VRF custA che per VRF custB.

In questa schermata viene mostrata la tabella di routing per il "server" dell'istanza VRF del servizio condiviso:

```
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: per la rete di destinazione è presente un solo percorso dalla prospettiva del router PE in uscita (**iguana**).

Pertanto, non è stato possibile distinguere il traffico proveniente da più VPN del cliente e il traffico di ritorno non è in grado di raggiungere la VPN appropriata. **Nel caso in cui il servizio condiviso deve essere definito come istanza VRF, la funzione NAT deve essere spostata nella PE in entrata.**

NAT PE in ingresso

Nell'esempio, i router di confine del provider contrassegnati come **gila** e **dragon** sono configurati per NAT. Viene definito un pool NAT per ogni VPN del cliente collegato che deve accedere al servizio condiviso. Il pool appropriato viene utilizzato per ogni indirizzo di rete NAT del cliente. Il protocollo NAT viene eseguito solo sui pacchetti destinati all'host del servizio condiviso in modalità 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: in questo scenario i pool condivisi non sono supportati. Se la LAN del servizio condiviso (in uscita PE) è connessa tramite un'interfaccia generica, il pool NAT può essere condiviso.

Un ping originato da un indirizzo duplicato (172.31.1.1) all'interno di ognuna delle reti collegate a **neuse** e **capeSONY8** restituisce le seguenti voci NAT:

Da **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: lo stesso indirizzo locale interno (172.31.1.1) viene convertito in ognuno dei pool definiti in base al VRF di origine. Il VRF può essere visualizzato nel comando **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
```

Queste visualizzazioni mostrano le informazioni di routing per ciascuna delle VPN collegate localmente per il cliente A e il 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: è stato aggiunto un percorso per ciascun pool NAT dalla configurazione statica. Queste subnet vengono successivamente importate nel VRF del server condiviso all'iguana del router PE in uscita:

```
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
```

Configurazioni

Alcune informazioni estranee sono state rimosse dalle configurazioni per brevità.

```
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: le interfacce che si trovano di fronte alle reti del cliente sono designate come interfacce "interne" NAT e le interfacce MPLS come interfacce "esterne" NAT.

```
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
```

Il **drago** del router avrebbe una configurazione molto simile a **gila**.

Pacchetti in arrivo al PE centrale dopo il NAT PE in ingresso

Le tracce seguenti illustrano i requisiti per i pool NAT univoci quando la rete di servizi condivisi di destinazione è configurata come istanza VRF. Fare nuovamente riferimento allo schema della Figura 5. I pacchetti mostrati di seguito sono stati acquisiti mentre entravano nell'interfaccia IP

MPLS e1/0/5 sul router **iguana**.

Eco dal cliente A VPN

Qui vediamo una richiesta echo proveniente dall'indirizzo IP di origine 172.31.1.1 in VRF custA. L'indirizzo di origine è stato tradotto in 192.168.1.1 come specificato dalla configurazione 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 da VPN cliente B

Qui vediamo una richiesta echo proveniente dall'indirizzo IP di origine 172.31.1.1 in VRF custB. L'indirizzo di origine è stato tradotto in 192.168.2.1 come specificato dalla configurazione 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: il valore dell'etichetta MPLS è *0019* in entrambi i pacchetti mostrati sopra.

Risposta Echo al cliente A VPN

Successivamente, viene visualizzata una risposta echo per tornare all'indirizzo IP di destinazione 192.168.1.1 nella appliance VRF CustA. L'indirizzo di destinazione viene convertito in 172.31.1.1 dalla funzione NAT PE in entrata.

```
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:
```

Risposta di Echo al cliente B VPN

Qui vediamo una risposta echo che torna all'indirizzo IP di destinazione 192.168.1.1 in VRF custB. L'indirizzo di destinazione viene convertito in 172.31.1.1 dalla funzione NAT PE in entrata.

```
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: nei pacchetti restituiti, i valori delle etichette MPLS sono inclusi e differiscono: *001A* per VRF custA e *001D* per VRF custB.

Echo dal cliente A VPN - La destinazione è un'interfaccia generica

Questo gruppo di pacchetti successivo mostra la differenza quando l'interfaccia della LAN a servizio condiviso è un'interfaccia generica e non fa parte di un'istanza VRF. La configurazione è stata modificata per utilizzare un pool comune per entrambe le VPN locali con indirizzi IP sovrapposti.

```
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".]
```

Echo dal cliente B VPN - La destinazione è un'interfaccia generica

Qui vediamo una richiesta echo proveniente dall'indirizzo IP di origine 172.31.1.1 in VRF custB. L'indirizzo di origine è stato convertito in 192.168.1.3 (dal pool comune SSPOOL1) come specificato dalla configurazione 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 l'interfaccia in uscita PE è un'interfaccia generica (non un'istanza VRF), le etichette imposte sono diverse. In questo caso, *0x19* e *0x1F*.

Echo Risposta al cliente A VPN - La destinazione è un'interfaccia generica

Successivamente, viene visualizzata una risposta echo per tornare all'indirizzo IP di destinazione 192.168.1.1 nella appliance VRF CustA. L'indirizzo di destinazione viene convertito in 172.31.1.1 dalla funzione NAT PE in entrata.

```
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".]
```

Risposta di Echo al cliente B VPN - La destinazione è un'interfaccia generica

Qui vediamo una risposta echo che torna all'indirizzo IP di destinazione 192.168.1.3 in VRF custB.

L'indirizzo di destinazione viene convertito in 172.31.1.1 dalla funzione NAT PE in entrata.

```
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: poiché le risposte sono destinate a un indirizzo globale, non vengono imposte etichette VRF.

Se l'interfaccia di uscita dal segmento LAN del servizio condiviso è definita come interfaccia generica, è consentito un pool comune. Il ping ha come risultato le seguenti voci NAT nel **gila del** router:

```
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
```

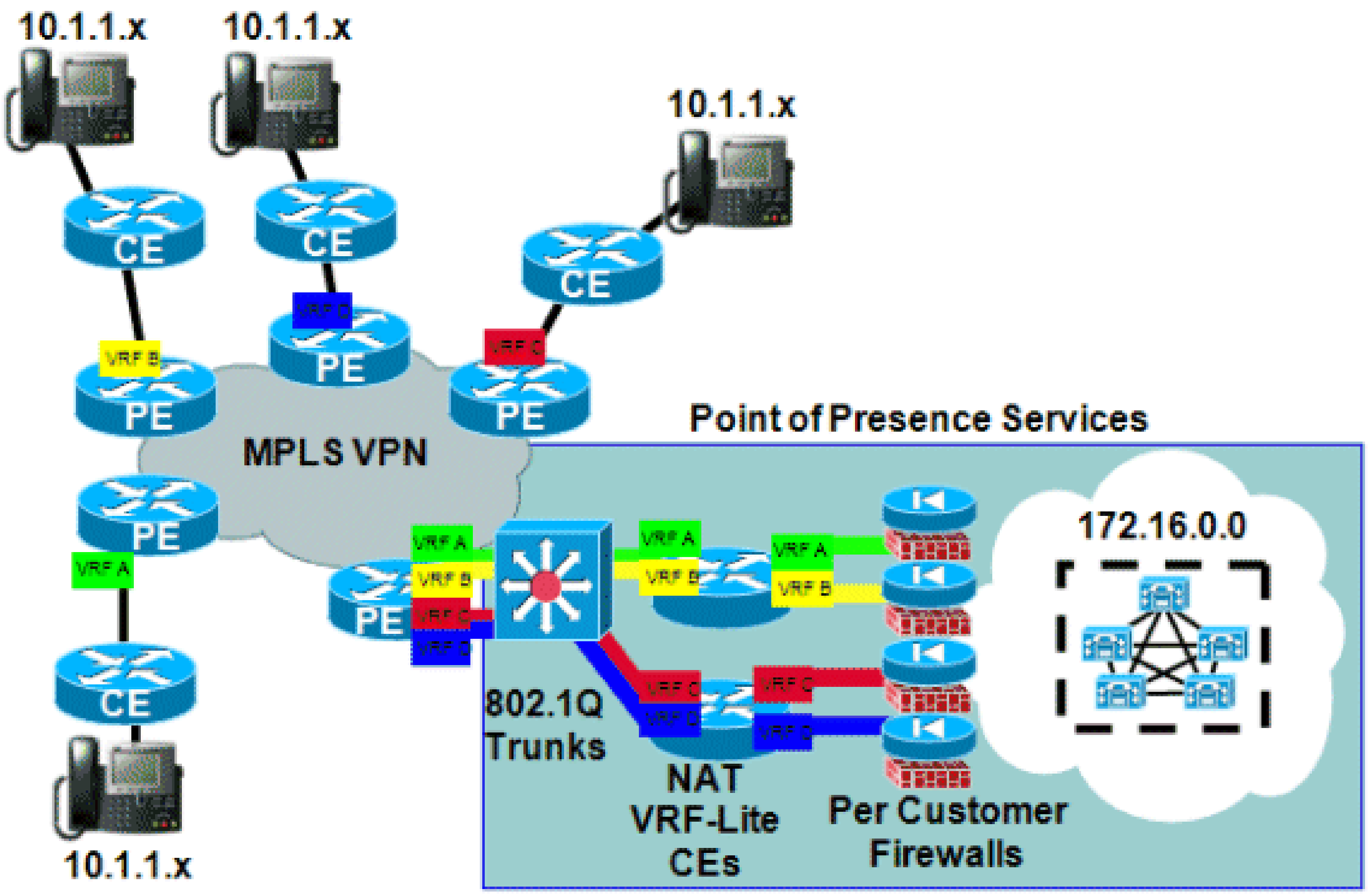
```
.Jan  2 09:35:12 EST: NAT-ip2tag: Punting to process
.Jan  2 09:35:12 EST: NAT-ip2tag : Tag Pkt s=172.31.1.1, d=88.1.88.8, vrf=custA
.Jan  2 09:35:12 EST: NAT-ip2tag: Punting to process
.Jan  2 09:35:12 EST: NAT-ip2tag : Tag Pkt s=172.31.1.1, d=88.1.88.8, vrf=custA
.Jan  2 09:35:12 EST: NAT-ip2tag: 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#
```

Esempio di servizio

La Figura 8 mostra un esempio di servizio PBX IP virtuale condiviso. Illustra una variante degli esempi in entrata e in uscita descritti in precedenza.

In questo progetto, il servizio VoIP condiviso è front-end da un set di router che eseguono la funzione NAT. Questi router dispongono di più interfacce VRF che utilizzano una funzione nota come VRF-Lite. Il traffico viene quindi indirizzato al cluster Cisco CallManager condiviso. I servizi firewall vengono forniti anche per singola azienda. Le chiamate interaziendali devono passare attraverso il firewall, mentre quelle interaziendali vengono gestite attraverso la VPN del cliente utilizzando lo schema di indirizzamento interno dell'azienda.

Figura 8: Esempio di servizio PBX virtuale gestito



Disponibilità

Il supporto NAT di Cisco IOS per le VPN MPLS è disponibile nella versione 12.2(13)T di Cisco IOS ed è disponibile per tutte le piattaforme che supportano MPLS e possono eseguire questa procedura di rilascio anticipato.

Conclusioni

Cisco IOS NAT dispone di funzionalità che consentono l'installazione scalabile dei servizi condivisi. Cisco continua a sviluppare il supporto ALG (NAT Application Level Gateway) per i protocolli importanti per i clienti. Miglioramenti delle prestazioni e accelerazione hardware per le funzioni di traduzione garantiranno che NAT e ALG forniscano soluzioni accettabili per un certo periodo di tempo. Cisco segue tutte le attività di normazione rilevanti e le azioni della community. Con lo sviluppo di altri standard, il loro utilizzo verrà valutato in base alle esigenze, ai requisiti e all'applicazione del cliente.

Informazioni correlate

- Cisco IOS NAT Application Layer Gateway
- Architetture MPLS e VPN
- Progettazione e implementazione avanzate di MPLS
- Documentazione e supporto tecnico – Cisco Systems