

Introduzione a VRF-LITE

Le VPN rivestono un ruolo di fondamentale importanza nelle reti di comunicazione dati. Le aziende hanno necessita' di collegare le loro sedi con dei circuiti sicuri, che impediscano intrusioni e violazioni della sicurezza. Per preservare i propri dati, le aziende hanno negli anni '80 e negli anni '90 utilizzato i servizi che mettevano a disposizione i carrier dell'epoca, quali circuiti X.25, Frame-relay e poi ATM. Questo significava che in ogni sede aziendale era necessario disporre di costosi apparecchi in grado di supportare questi protocolli. Gli operatori, a loro volta, dovevano avere una rete a copertura geografica sufficiente per fornire questa tipologia di servizi.

In Italia l'operatore monopolista possedeva una rete X.25, successivamente affiancata da una rete Frame-Relay e quindi da una piu' recente rete ATM molto nota, oggi giorno, perche' rete di raccolta del traffico che viaggia nei collegamenti XDSL.

In un mondo oramai dominato dal protocollo TCP/IP e dall'ethernet gli svantaggi derivanti dall'uso di una tale infrastruttura appaiono evidenti. L'uso del protocollo IP infatti ha spostato l'uso dei circuiti X.25, Frame-relay o ATM solo come protocolli di rete, delegando al TCP/IP il vero e proprio trasporto dei dati. Viene allora spontaneo chiedersi perche' utilizzare ancora gli obsoleti X.25 e Frame-relay, o il costoso ATM, e non tecnologie piu' efficienti e moderne.

Con l'abbassamento dei costi e l'aumento delle performance, e l'arrivo di piu' economici collegamenti di tipo XDSL, molte VPN si sono realizzate a livello 3, utilizzando protocolli quali IPSEC, svincolate dai carrier. In questo caso la VPN e' ritagliata su un collegamento internet, senza l'intervento del provider.

D'altro canto per i fornitori di servizio non vi era una soluzione differente dall'uso di protocollo Frame-relay o ATM per fornire VPN. Poiche' alla fine i clienti utilizzavano il TCP/IP nel caso di VPN complesse era necessario configurare un gran numero di circuiti e inserire nei router del cliente un notevole numero di policy. La necessita' era quella di poter gestire le VPN tramite il protocollo IP svincolandosi il prima possibile dalla commutazione di circuito. E' conservarne l'amministrazione lato provider, semplificando il piu' possibile i costi lato cliente.

L'avvento del protocollo MPLS ha fornito ai provider una soluzione ai loro problemi consentendo loro di svincolarsi dalle vecchie pile protocollari e di concentrarsi sull'uso di IP, ethernet, SDH e XDSL. MPLS e' una tecnica di trasmissione che utilizza delle etichette per differenziare i vari flussi di traffico. Consente di utilizzare il protocollo IP con tutte le specifiche di qualita' di servizio, ingegneria di traffico e gestione VPN mancanti nella pila protocollare TCP/IP. Un ultimo vantaggio di Frame-relay e ATM era la possibilita' di gestire la qualita' del servizio e l'ingegneria del traffico in modo preciso, evitando congestioni, e garantendo ai clienti la banda contrattualizzata.

Questo documento in realta' non parla di MPLS, ma di un suo "prodotto derivato", ovvero il VRF-lite. MPLS viene infatti utilizzato in ambito campus o provider. Ma questa tecnologia supporta cosi' bene le VPN che si e' sviluppato VRF-lite, un sottoinsieme di VRF, utilizzato invece nelle VPN/MPLS. VRF significa "Virtual Routing and Forwarding" ed e' la possibilita' di creare un router virtuale per ogni singola VPN e dei servizi ad essa correlati che, quando supportati da VRF, vengono indicati come "VRF-aware".

Pertanto in questo documento si espone il VRF-lite. Quindi non troverete configurazioni MPLS. In ogni caso difficilmente troverete MPLS in una rete aziendale, dove molto piu' facilmente potrete trovare o usare il VRF-lite. Nella rete di un provider difficilmente troverete VRF-lite e molto piu' comunemente troverete MPLS.

Con il Virtual Routing and Forwarding (VRF) si crea un router logico a tutti gli effetti. Un VRF e' costituito da una tabella di routing IP, da alcune interfacce, e da alcuni servizi ad esso associati (DHCP, NAT etc.). Possono essere configurati piu' router logici in uno stesso router fisico.

Con questo sistema e' possibile gestire in uno stesso router piu' circuiti logici separati a livello 3. Possiamo gestire piu' VPN anche con overlapping degli indirizzi IP in quanto appartenenti a schemi logici isolati tra di loro.

VRF-lite e' una semplificazione di VRF. Si usa a livello locale e non a livello provider dove si hanno le MPLS-VPN. I protocolli di routing RIP, EIGRP, OSPF e BGP sono supportati. VRF-lite e' a disposizione nell'immagine IOS di default di molti router CISCO, a partire dalla serie 8XX.

Nota sugli esempi: alcuni esempi potrebbero non funzionare se non avete un IOS abbastanza recente o con features adeguate. Conviene sicuramente utilizzare almeno IOS 12.3 o 12.4. Con alcuni IOS alcune funzionalita' VRF-lite potrebbero funzionare solo parzialmente o potrebbero non essere supportati alcuni protocolli di routing.

Prerequisiti: anche se questo documento e' una introduzione al VRF-lite prerequisito fondamentale e' una buona conoscenza delle problematiche di routing in ambiente LAN/WAN, con i vari protocolli di routing e cosi' via. Il documento non ha come obiettivo la spiegazione dei comandi di configurazione dei router Cisco in una configurazione non VRF, che vengono dati per scontati.

Configurazione 1 - Configurazione iniziale -

```
interface FastEthernet0/0
  description --- WAN ---
  ip address 192.168.30.124 255.255.255.0
!
interface FastEthernet0/1
  description --- LAN ---
  ip address 192.168.1.2 255.255.255.0
!
ip route 0.0.0.0 0.0.0.0 192.168.30.254 <-- INTERNET (WAN)
ip route 192.168.10.0 255.255.255.0 192.168.1.1 <-- LAN
```

Descrizione

In questa configurazione non e' presente VRF-lite. E' una configurazione di partenza. Abbiamo una unica tabella di routing.

```
ROUTER_A#show ip route
...snip...
Gateway of last resort is 192.168.30.254 to network 0.0.0.0

C 192.168.30.0/24 is directly connected, FastEthernet0/0
S 192.168.10.0/24 [1/0] via 192.168.1.1
C 192.168.1.0/24 is directly connected, FastEthernet0/1
S* 0.0.0.0/0 [1/0] via 192.168.30.254
```

Configurazione 2 - Creiamo il VRF -

```

ip vrf blue      (1)
  rd 100:10     (2)          <--- Identificativo tabella routing. La notazione standard
e' AS:valore o IP:valore
!
!
interface FastEthernet0/0
description --- WAN ---
ip address 192.168.30.124 255.255.255.0
!
interface FastEthernet0/1
description --- LAN ---
ip address 192.168.1.2 255.255.255.0
!
!
ip classless
ip route 0.0.0.0 0.0.0.0 192.168.30.254
ip route 192.168.10.0 255.255.255.0 192.168.1.1
!

```

Descrizione

Osserviamo le righe (1) (2). Si definisce una nuova tabella di routing di nome "blue" e con identificativo 100:10. La nuova tabella di routing al momento e' vuota, infatti le righe di routing statico configurate fanno riferimento alla tabella di routing "globale" ovvero quella di default. L'identificativo, chiamato route-distinguisher (rd), viene attaccato agli indirizzi IP come prefisso, per distinguere indirizzi uguali ma appartenenti a VRF differenti ovvero a tabelle di routing differenti. Questo pero' sara' utile solo con l'uso di BGP. Fino ad allora potremo configurare un'istanza VRF con la sola riga (1). Come ci aspettiamo la tabella di routing "blue" e' vuota:

```
ROUTER_A#show ip route vrf blue
```

```

Routing Table: blue
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, su - IS-IS summary, 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

```

La tabella di routing globale invece contiene tutte le righe di routing statico:

```
ROUTER_A#show ip route
```

```

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, su - IS-IS summary, 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 192.168.30.254 to network 0.0.0.0

C 192.168.30.0/24 is directly connected, FastEthernet0/0
S 192.168.10.0/24 [1/0] via 192.168.1.1
C 192.168.1.0/24 is directly connected, FastEthernet0/1
S* 0.0.0.0/0 [1/0] via 192.168.30.254

```

Configurazione 3 - Popoliamo il VRF -

ROUTER_A

```

!
ip vrf blue
rd 100:10
!
!
interface FastEthernet0/0
description --- RETE WAN ---
ip address 192.168.30.124 255.255.255.0
!
interface FastEthernet0/1
description --- RETE LAN ---
ip vrf forwarding blue          <--- osservate questa riga (1)
ip address 192.168.1.2 255.255.255.0
!
ip classless
ip route 0.0.0.0 0.0.0.0 192.168.30.254
ip route 192.168.10.0 255.255.255.0 192.168.1.1
ip route vrf blue 192.168.50.0 255.255.255.0 192.168.1.1    <--- osservate questa riga
(2)
!
!

```

Descrizione

Con la riga (1) inseriamo l'interfaccia Fastethernet0/1 nel vrf "blue". Questo vuol dire che l'interfaccia fara' parte del VRF "blue". E' come se facesse parte di un'altro router, o per analogia con uno switch, ad una VLAN differente. Non apparira' piu' in "show ip route" in quanto facente parte del VRF "blue". Abbiamo cosi' creato un nuovo router virtuale con una interfaccia e una sua tabella di routing. Con la riga (2) aggiungiamo una route statica al VRF "blue" :

```

ROUTER_A#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
Gateway of last resort is 192.168.30.254 to network 0.0.0.0

```

```

C 192.168.30.0/24 is directly connected, FastEthernet0/0
S* 0.0.0.0/0 [1/0] via 192.168.30.254

```

```

ROUTER_A# show ip route vrf blue

```

```

Routing Table: blue
Gateway of last resort is not set

```

```

S 192.168.50.0/24 [1/0] via 192.168.1.1
C 192.168.1.0/24 is directly connected, FastEthernet0/1
ROUTER_ESTERNO#

```



A questo punto siamo in grado di configurare piu' processi di routing e di associare ad essi delle interfacce e delle righe di routing statico. Stiamo facendo, per il layer 3, qualcosa di simile a quanto si puo' fare con uno switch a livello 2, creando delle VLAN.

Nel caso di reti con piu' router e' necessario trasportare i diversi flussi di traffico tra gli apparati, mantenendoli pero' isolati. Per seguire l'analogia con gli switch vorremmo avere l'equivalente di un trunk 802.1q ovvero tenere i VRF separati ma metterli in comunicazione con gli equivalenti VRF di altri router in un ambiente LAN/WAN.

Dimentichiamo l'analogia con gli switch, che usata alla lunga puo' generare confusione, e proseguiamo con gli esempi.



Configurazione 4 - Comandi utili -

```
!  
ip vrf bianco  
rd 100:10  
!  
ip vrf blue  
rd 102:10  
!  
ip vrf nero  
rd 101:10  
!  
!  
interface Loopback0  
ip vrf forwarding bianco  
ip address 192.168.88.1 255.255.255.0  
!  
interface Loopback1  
ip vrf forwarding nero  
ip address 192.168.89.1 255.255.255.0  
!  
interface Loopback2  
ip vrf forwarding blue  
ip address 192.168.90.1 255.255.255.0  
!  
interface Ethernet0/0  
ip address 192.168.1.2 255.255.255.252  
!  
ip classless  
ip route 192.168.78.0 255.255.255.0 192.168.1.1  
ip route 192.168.79.0 255.255.255.0 192.168.1.1  
ip route 192.168.80.0 255.255.255.0 192.168.1.1
```

Descrizione

Nell'esempio abbiamo tre VRF e tre interfacce associate ad essi. Osserviamo i seguenti comandi:

Per avere un dettaglio dei VRF configurati:

```
ROUTER_A#sh ip vrf detail  
VRF bianco; default RD 100:10; default VPNID <not set>  
Interfaces:  
Loopback0  
Connected addresses are not in global routing table  
No Export VPN route-target communities  
No Import VPN route-target communities  
No import route-map  
No export route-map  
VRF blue; default RD 102:10; default VPNID <not set>  
Interfaces:  
Loopback2  
Connected addresses are not in global routing table  
No Export VPN route-target communities  
No Import VPN route-target communities  
No import route-map  
No export route-map  
VRF nero; default RD 101:10; default VPNID <not set>  
Interfaces:  
Loopback1  
Connected addresses are not in global routing table  
No Export VPN route-target communities  
No Import VPN route-target communities  
No import route-map  
No export route-map
```

Attenzione al fatto che il comando "ping" fa riferimento alla tabella di routing di default (che da ora in poi chiameremo "globale"), quindi:

```
ROUTER_A#ping 192.168.89.1
```

```
Type escape sequence to abort.  
Sending 5, 100-byte ICMP Echos to 192.168.89.1, timeout is 2 seconds:  
.....  
Success rate is 0 percent (0/5)
```

fallisce pur essendo 192.168.89.1 un indirizzo IP di una interfaccia nello stesso router ma appartenente ad un VRF diverso. Invece:

```
ROUTER_A#ping vrf bianco 192.168.88.1
```

```
Type escape sequence to abort.  
Sending 5, 100-byte ICMP Echos to 192.168.88.1, timeout is 2 seconds:  
!!!!!!  
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms
```

Stessa cosa dicasi per gli altri comandi. Ad esempio:

```
ROUTER_A#show ip route 192.168.88.1  
% Network not in table
```

fallisce anche se la riga di routing esiste, in quando associata ad interfaccia direttamente connessa, ma agganciata ad un diverso VRF. Pertanto:

```
ROUTER_A#show ip route vrf bianco 192.168.88.1  
Routing entry for 192.168.88.0/24  
Known via "connected", distance 0, metric 0 (connected, via interface)  
Routing Descriptor Blocks:  
* directly connected, via Loopback0  
Route metric is 0, traffic share count is 1
```



Configurazione 5 - Accesso ad internet parte I -

Supponiamo di avere piu' VRF e voler dare l'accesso ad Internet a tutti mantenendo pero' i VRF isolati tra loro. L'interfaccia verso internet normalmente non ha VRF e come conseguenza nessuna VPN/VRF avra' accesso ad internet in quanto i VRF non possono utilizzare le route della tabella globale. Inserire una riga di routing di default in ogni VRF non servira' a nulla in quanto l'interfaccia verso internet e' sempre nella tabella di routing globale:

ROUTER_A

```
interface Ethernet0/0
ip address 192.168.1.2 255.255.255.252
...
ip route vrf bianco 0.0.0.0 0.0.0.0 192.168.1.1
...
```

```
ROUTER_A#show ip route vrf bianco
Codes: C - connected, S - static, I - IGRP, 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, E - EGP
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

```
C 192.168.88.0/24 is directly connected, Loopback0          <--- notate l'assenza della
route statica
ROUTER_A#
ROUTER_A#
```

La soluzione consiste nell'utilizzare la parola chiave "global" nella route statica. Questa indica che il next-hop si trova nella tabella di routing globale:

```
interface Ethernet0/0
ip address 192.168.1.2 255.255.255.252
...
ip route vrf bianco 0.0.0.0 0.0.0.0 192.168.1.1 global
```

```
ROUTER_A# sh ip route vrf bianco
Codes: C - connected, S - static, I - IGRP, 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, E - EGP
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 192.168.1.1 to network 0.0.0.0

```
C 192.168.88.0/24 is directly connected, Loopback0
S* 0.0.0.0/0 [1/0] via 192.168.1.1          <--- notate come la riga di route sia stata recepita
                                             anche se eth0/0 non e' nel vrf bianco
```

I differenti VRF possono attingere il next-hop dalla tabella di routing globale ma non tra di loro (se non con BGP)

Configurazione 6 - Accesso ad internet con NAT parte II -

La configurazione mostrata nel punto 5 ha senso solo nel caso in cui si utilizzino ip pubblici. Solo in quel caso la navigazione internet funzionerebbe correttamente. In tutti gli altri casi, ovvero con il tradizionale uso di ip privati, sarà necessario utilizzare il NAT. Il NAT virtual interface, o NVI, è l'adattamento del NAT per l'uso in ambiente VRF. Si potranno infatti configurare diversi profili di NAT per differenti VRF, indipendenti tra di loro. Possiamo allora dire che il NAT è "VRF-aware". Osservate questa configurazione:

```
!
ip vrf bianco
rd 100:10
!
ip vrf nero
rd 101:10
!
interface ATM0
no ip address
no atm ilmi-keepalive
dsl operating-mode auto
!
interface ATM0.1 point-to-point
ip address 12.123.141.12 255.255.255.0
ip nat enable
ip virtual-reassembly
pvc 8/35
encapsulation aal5snap
!
interface ethernet0
description --- LAN1 ---
ip vrf forwarding nero
ip address 192.168.30.2 255.255.255.0          (1)
ip nat enable
ip virtual-reassembly
!
interface ethernet1
description --- LAN2 ---
ip vrf forwarding bianco
ip address 192.168.30.2 255.255.255.0          (2)

ip route 0.0.0.0 0.0.0.0 ATM0.1                (5)
ip route vrf nero 0.0.0.0 0.0.0.0 12.123.141.12 global (4)
!
ip nat pool TEST 12.123.141.12 12.123.141.12 netmask 255.255.255.0
ip nat source list 20 pool TEST vrf nero overload (3)
!
access-list 20 permit 192.168.30.0 0.0.0.255
!
```

Si tratta di una semplice LAN con un router dotato di interfaccia ADSL e due ethernet. Le due ethernet le possiamo supporre collegate a due LAN di due aziende differenti, che condividono lo stesso router per l'accesso ad internet ma devono restare isolate tra di loro.

Con una configurazione tradizionale avremmo dovuto separare il traffico tra le due reti con delle access-list, e oltretutto avremmo dovuto fare attenzione all'indirizzamento adottato, in quanto (senza VRF) le due reti devono avere due classi di IP private differenti.

Utilizzando VRF-lite la configurazione si semplifica ed è molto più semplice da amministrare, inoltre le due reti sono completamente indipendenti, infatti notate l'uso della stessa classe di IP privati da parte delle due LAN. Nessun conflitto, in quanto vi sono due VRF differenti. La LAN2 non ha servizio di navigazione internet.

Con NVI si utilizza il comando "ip nat enable" invece dei più noti "ip nat inside", "ip nat outside". La riga chiave è la (3) dove si aggancia il pool degli indirizzi al vrf specifico.

Con questa configurazione solo la LAN1 navighera' su internet. Notate come al punto (4) si sia utilizzato l'ip 12.123.141.12 invece dell'interfaccia atm0.1 come next-hop. Con i VRF non e' possibile specificare una interfaccia fisica come next-hop. Questo rende necessario la riga (5) per il raggiungimento dell'interfaccia fisica.

Ricordiamoci che, operando con i VRF, ogni comando dev'essere riferito al VRF in oggetto, anche per il NAT:

```
show ip nat trans vrf bianco
debug ip nat vrf
clear ip nat trans vrf
```

Negli esempi presentati abbiamo configurato VRF-LITE e spiegato come sia possibile creare diverse tabelle di routing isolate tra di loro all'interno di un unico router fisico. Associando le interfacce ai VRF, fisiche o logiche, e' possibile isolare gruppi di utenti. Possiamo inserire in un VRF un tunnel oppure una subinterfaccia ethernet. Siamo insomma in grado di creare gruppi isolati di utenti. Siamo anche in grado di fare navigare su internet alcuni di questi gruppi o tutti.

Cio' che vorremmo ottenere e' la possibilita' di decidere se, e entro quali limiti, questi differenti VRF possono importare o esportare righe di routing consentendo una comunicazione tra VLAN, molto utile in reti complesse.

Bisogna tenere in mente che VRF-Lite nasce per fare traffic-isolation, e per la raccolta di traffico verso un provider BGP/MPLS. Tuttavia, sempre restando al di fuori delle VPN/MPLS possiamo utilizzare il BGP anche su un singolo router, solo per gestire il traffico tra diverse VLAN. Restiamo con un singolo router. Segue una coraggiosa configurazione fatta con un semplice Cisco 877 e IOS Service Provider 12.4 (che supporta il BGP).

Configurazione 7 - Routing BGP e comunicazione tra le VPN –

Non avrei mai immaginato di configurare il protocollo BGP su un unico router, senza neighbors, ottenendo una configurazione interessante ed utilizzabile, anzi chiave per spingere al massimo l'uso del VRF-lite. Nell'esempio vediamo per la prima volta l'uso del comando "route-target". I route-target (RT) sono degli identificativi associati ad un VRF che indicano la disponibilita' di esportare o importare righe di routing verso un differente VRF. Affinche' due VRF comunichino tra di loro sara' necessario che ognuno di loro esporti la propria tabella di routing e che l'altro la importi. I valori numerici indicati negli RT si chiamano "extended communities". Questi identificativi vengono usati dal protocollo BGP per il corretto import/export tra VRF. Il motivo dell'uso del protocollo BGP e' che e' l'unico in grado di agganciare alle route delle informazioni aggiuntive. Le "communities" infatti non sono una novita' in ambiente BGP e sono usate correntemente in ambienti non VRF.

```

!
!
ip vrf bianco
rd 1:1
route-target export 150:150
route-target import 101:10
!
ip vrf nero
rd 101:10
route-target export 101:10
route-target import 150:150
!
!
interface Loopback1
ip vrf forwarding bianco
ip address 192.168.30.2 255.255.255.0
!
interface Loopback2
ip vrf forwarding nero
ip address 192.168.31.2 255.255.255.0
!
...snip...
!
interface Vlan1
description --- WAN ---
ip vrf forwarding bianco
ip address 192.168.1.1 255.255.255.252
ip virtual-reassembly
!
router bgp 1000
no synchronization
bgp router-id 1.1.1.1
bgp log-neighbor-changes
no auto-summary
!
address-family ipv4 vrf nero
no synchronization
network 192.168.31.0
exit-address-family
!
address-family ipv4 vrf bianco
no synchronization
network 192.168.1.0
network 192.168.30.0
exit-address-family
!

```

<---- il vrf bianco esporta le sue righe di routing con l'extended community 150:150
<---- il vrf bianco importa le righe di routing con l'extended community 101:10, del vrf nero

<---- vrf nero esporta le sue righe di routing
<---- il vrf nero importa l'extended community 150:150 garantendosi l'accesso al VRF bianco

Importante e' osservare come nella tabella di routing del VRF nero vi sia anche la rete 30.0 appartenente al VRF-bianco

```
Router#show ip route vrf nero
```

```
C 192.168.31.0/24 is directly connected, Loopback2
B 192.168.30.0/24 is directly connected, 00:31:21, Loopback1
```

Nella tabella del VRF bianco vi e' anche la rete 31.0, del VRF nero

```
Router#show ip route vrf bianco
```

```
B 192.168.31.0/24 is directly connected, 00:01:46, Loopback2
C 192.168.30.0/24 is directly connected, Loopback1
  192.168.1.0/30 is subnetted, 1 subnets
C 192.168.1.0 is directly connected, Vlan1
```

Il risultato e' che dal VRF bianco si raggiunge l'indirizzo 192.168.31.2, del VRF nero, e viceversa

```
Router#ping vrf bianco 192.168.31.2
```

```
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.31.2, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/4 ms
```

```
Router#ping vrf nero 192.168.30.2
```

```
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.30.2, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/4 ms
```

Vediamo quali route sono contenute nella tabella BGP:

```
Router#show ip bgp vpnv4 all
```

```
...
Network                Next Hop Metric LocPrf Weight Path
Route Distinguisher: 1:1 (default for vrf bianco)
*> 192.168.1.0/30        0.0.0.0    0 32768 ?
*> 192.168.30.0         0.0.0.0    0 32768 i
*> 192.168.31.0         0.0.0.0    0 32768 i
Route Distinguisher: 101:10 (default for vrf nero)
*> 192.168.1.0/30      0.0.0.0    0 32768 ?
*> 192.168.30.0        0.0.0.0    0 32768 i
*> 192.168.31.0        0.0.0.0    0 32768 i
Router#
```

Senza l'uso di MPLS abbiamo raggiunto il nostro obiettivo che ci consente di creare VPN complesse. Immaginiamo infatti il caso in cui un'azienda ha tre sedi A, B, C. Sia A che B devono raggiungere C ma non devono raggiungersi a vicenda. Bastera' allora utilizzare tre RD differenti per le tre sedi e con il BGP fare l'import/export tra A-C e B-C ma non A-B.

I protocolli di routing con VRF-lite sono pensati per un uso lato CE (customer-edge) quando ci si aggancia da un PE (provider-edge) per ottenere servizi MPLS. Lato provider ci sarà MPLS/BGP. Lato cliente si potrà utilizzare RIP, EIGRP, OSPF oppure BGP. In ogni caso lato provider le route verranno redistribuite nel BGP della rete MPLS dell'operatore. Ma noi vogliamo restare in ambito VRF-lite. Vediamo come utilizzare allora i protocolli di routing per agganciare tra loro tabelle VRF di router distinti mantenendo l'isolamento delle VPN.

Configurazione 8 - Routing RIP lato CE -

RIP determina il VRF di una route in base al VRF dell'interfaccia di provenienza. RIP propaga le route sulle interfacce in cui è attivo e che hanno VRF corrispondente a quello in cui la route è definita. Vediamo una possibile configurazione CE-CE:

```
ip vrf bianco
rd 100:10
!
ip vrf nero
rd 101:10
!
!
interface Loopback1
ip vrf forwarding nero
ip address 192.168.89.1 255.255.255.0
!
interface Loopback3
ip vrf forwarding bianco
ip address 192.168.88.1 255.255.255.0
!
interface Tunnel1
ip vrf forwarding nero
ip address 192.168.40.2 255.255.255.252
tunnel source 192.168.1.2          <--- il tunnel attraversa un collegamento senza VRF. Avremmo usato il comando
"tunnel vrf NAME" in caso contrario
tunnel destination 192.168.1.1
!
interface Tunnel10
ip vrf forwarding bianco
ip address 192.168.50.2 255.255.255.252
tunnel source 192.168.1.2
tunnel destination 192.168.1.1
!
interface Ethernet0/0
ip address 192.168.1.2 255.255.255.0
half-duplex
!
router rip
version 2
!
address-family ipv4 vrf nero
network 192.168.40.0
network 192.168.89.0
no auto-summary
version 2
exit-address-family
!
address-family ipv4 vrf bianco
network 192.168.50.0
```

```
network 192.168.88.0
no auto-summary
version 2
exit-address-family
!
```

Questo esempio e' stato reso volutamente piu' complicato con l'uso dei tunnel. Il motivo e' che per tenere i VRF separati tra router e router abbiamo bisogno di una interfaccia per ogni VRF. Se non vi sono interfacce fisiche a sufficienza dobbiamo crearne di logiche. Quindi possiamo usare tunnel o, se preferite, subinterfacce ethernet laddove possibile. In questo esempio gli aggiornamenti RIP passano attraverso i tunnel. I tunnel GRE non sono certo una novita' in ambiente Cisco. Ma una loro applicazione relativamente recente li vede accoppiati alla feature VRF-lite.

Il comando "address-family ipv4 vrf NOME" permette di gestire i diversi VRF in modo disgiunto. Attenzione al fatto che RIP perde le informazioni relative all'RD oppure ad eventuali RT. Infatti non e' in grado di trasportarle. La loro configurazione in questi esempi e' utile solo laddove c'e' il BGP, in grado di trasportarli e gestirli.

Supponiamo di avere una LAN di tipo Campus dove bisogna distribuire vari livelli di accesso. Fino ad ora questo era fatto con le VLAN. Una VLAN guest darebbe un accesso ad Internet libero senza autenticazione mentre il traffico protetto andrebbe su VLAN differenti.

Utilizzando VRF si hanno nuove opportunita'. Un modo piu' semplice e' quello di raggruppare tutto il traffico della rete guest in un unico VRF in ciascuno switch di distribuzione. Quindi il traffico e trasportato attraverso la rete LAN mediante un GRE tunnel verso un dispositivo centrale che conduce verso internet.



Configurazione 9 - EIGRP –

A differenza del RIP l'EIGRP fornisce un migliore supporto in quanto e' in grado di gestire l'autonomous system. Con lo stesso criterio di configurazione del RIP e' pero' possibile associare un differente autonomous-system (AS) ad ogni VRF.

```

hostname yourname
!
ip vrf blue          <-- puo' essere diverso nei due router
rd 100:10           <-- puo' essere diverso nei due router
!
!
interface FastEthernet0/0
description -- back to back ---
ip vrf forwarding blue  <-- puo' essere diverso nei due router
ip address 192.168.1.2 255.255.255.0
duplex auto
speed auto
!
interface FastEthernet0/1
description --- internet ---
ip vrf forwarding blue  <-- puo' essere diverso nei due router
ip address 192.168.30.152 255.255.255.0
duplex auto
speed auto
!
router eigrp 100      <-- puo' essere diverso nei due router
auto-summary
!
address-family ipv4 vrf blue  <-- puo' essere diverso nei due router
network 192.168.1.0
auto-summary
autonomous-system 101  <-- dev'essere uguale al router neighbor per metter su la
                        sessione. Questo comando esiste solo se si specifica la
                        keyword vrf nel comando "address-family ipv4"
exit-address-family
!
ip route vrf blue 0.0.0.0 0.0.0.0 192.168.30.254
!

yourname#show ip eigrp vrf blue interfaces
IP-EIGRP interfaces for process 101

Xmit Queue Mean Pacing Time Multicast Pending
Interface Peers Un/Reliable SRTT Un/Reliable Flow Timer Routes
Fa0/0 1 0/0 9 0/1 50 0
yourname#

```

Descrizione

Con EIGRP possiamo agganciare due router distinti e trasferire le informazioni di routing dei corrispondenti VRF. Nelle configurazioni non e' necessario che i due (o piu') router condividano gli stessi rd. Cio' che aggancia i processi EIGRP di due router distinti e' l'autonomous system number. Ovviamente le "network" per essere propagate devono appartenere ad interfacce con VRF corrispondenti a quelli definiti con il comando address-family. L'autonomous system e' configurabile solo se si usa il comando "address-family".

Sfruttando diversi AS mettiamo in corrispondenza i processi EIGRP di router differenti. EIGRP non propaga l'RD tuttavia associando in fase di progetto gli RD agli AS possiamo mantenere separati i VRF tra router distinti. Ricordo ancora una volta che e' necessario BGP quando si vogliono propagare RD e RT.

Configurazione 10 - DHCP -

Anche il servizio DHCP e' VRF-aware ed e' quindi possibile attivare il DHCP esclusivamente nell'ambito di una singola interfaccia.

Descrizione

Supponiamo di scegliere la classe privata 192.168.152.0/24 da assegnare con DHCP. Innanzitutto e' fondamentale il comando "ip dhcp use vrf connected" che abilita il dhcp per le interfacce vrf direttamente connesse (ad es. Ethernet). Poi escludiamo l'ip 192.168.152.254, che e' il gateway, e altri 4 indirizzi IP che, in questo esempio, utilizziamo come ip statici nella network. Infine da notare il comando "vrf miovr" con il quale agganciamo il pool DHCP ad un ben preciso VRF.

```
ip dhcp use vrf connected
ip dhcp excluded-address 192.168.152.254
ip dhcp excluded-address 192.168.152.10 192.168.152.13
```

```
ip dhcp pool DHCPSEVER
  vrf miovr
  network 192.168.152.0 255.255.255.0
  dns-server 151.99.125.1 151.99.125.2
  default-router 192.168.152.254
```

```
ip vrf miovr
  rd 1055:52
```

Con il comando "show ip dhcp pool DHCPSEVER" possiamo vedere le statistiche sul pool che abbiamo configurato:

```
router#show ip dhcp pool DHCPSEVER
```

```
Pool DHCPSEVER :
  Utilization mark (high/low) : 100 / 0
  Subnet size (first/next) : 0 / 0
  VRF name : miovr
  Total addresses : 254
  Leased addresses : 1
  Pending event : none
  1 subnet is currently in the pool :
```

Current index	IP address range	Leased addresses
192.168.152.1	192.168.152.1 - 192.168.152.254	1

Ecco che arriva una richiesta (potete attivare il debug con "debug ip dhcp server event")

```
*Sep 17 14:19:52.088: DHCPD: Sending notification of DISCOVER:
*Sep 17 14:19:52.088: DHCPD: htype 1 chaddr 000e.3525.f394
*Sep 17 14:19:52.088: DHCPD: remote id 020a0000c0a898fe01000034
*Sep 17 14:19:52.088: DHCPD: circuit id 00000000
*Sep 17 14:19:52.088: DHCPD: table id 1 = vrf miovr
*Sep 17 14:19:52.088: DHCPD: Seeing if there is an internally specified pool class:
*Sep 17 14:19:52.088: DHCPD: htype 1 chaddr 000e.3525.f394
*Sep 17 14:19:52.088: DHCPD: remote id 020a0000c0a898fe01000034
*Sep 17 14:19:52.088: DHCPD: circuit id 00000000
*Sep 17 14:19:52.088: DHCPD: table id 1 = vrf miovr
*Sep 17 14:19:52.192: DHCPD: Sending notification of ASSIGNMENT:
*Sep 17 14:19:52.192: DHCPD: address 192.168.152.1 mask 255.255.255.0
*Sep 17 14:19:52.192: DHCPD: htype 1 chaddr 000e.3525.f394
*Sep 17 14:19:52.192: DHCPD: table id 1 = vrf miovr
```

*Sep 17 14:19:52.192: DHCPD: lease time remaining (secs) = 86400
*Sep 17 14:19:55.504: DHCPD: Sending notification of ASSIGNMENT:
*Sep 17 14:19:55.504: DHCPD: address 192.168.152.1 mask 255.255.255.0
*Sep 17 14:19:55.504: DHCPD: htype 1 chaddr 000e.3525.f394
*Sep 17 14:19:55.504: DHCPD: table id 1 = vrf miovr
*Sep 17 14:19:55.504: DHCPD: lease time remaining (secs) = 86400

Ecco che arriva una richiesta di rilascio dell'indirizzo:

*Sep 17 14:20:54.328: DHCPD: Sending notification of TERMINATION:
*Sep 17 14:20:54.328: DHCPD: address 192.168.152.1 mask 255.255.255.0
*Sep 17 14:20:54.328: DHCPD: reason flags: RELEASE
*Sep 17 14:20:54.328: DHCPD: htype 1 chaddr 000e.3525.f394
*Sep 17 14:20:54.328: DHCPD: table id 1 = vrf miovr
*Sep 17 14:20:54.328: DHCPD: lease time remaining (secs) = 86341
*Sep 17 14:20:54.328: DHCPD: returned 192.168.152.1 to address pool DHCPSEVER.
*Sep 17 14:20:57.744: DHCPD: Sending notification of DISCOVER:
*Sep 17 14:20:57.744: DHCPD: htype 1 chaddr 000e.3525.f394
*Sep 17 14:20:57.744: DHCPD: remote id 020a0000c0a898fe01000034
*Sep 17 14:20:57.744: DHCPD: circuit id 00000000
*Sep 17 14:20:57.744: DHCPD: table id 1 = vrf miovr
*Sep 17 14:20:57.744: DHCPD: Seeing if there is an internally specified pool class:
*Sep 17 14:20:57.744: DHCPD: htype 1 chaddr 000e.3525.f394
*Sep 17 14:20:57.744: DHCPD: remote id 020a0000c0a898fe01000034
*Sep 17 14:20:57.744: DHCPD: circuit id 00000000
*Sep 17 14:20:57.744: DHCPD: table id 1 = vrf miovr
*Sep 17 14:20:59.744: DHCPD: client requests 192.168.152.1.
*Sep 17 14:20:59.744: DHCPD: Adding binding to radix tree (192.168.152.1)
*Sep 17 14:20:59.744: DHCPD: VPN 'miovr'
*Sep 17 14:20:59.744: DHCPD: Adding binding to hash tree
*Sep 17 14:20:59.744: DHCPD: assigned IP address 192.168.152.1 to client 0100.0e35.25f3.94.
*Sep 17 14:20:59.784: DHCPD: Sending notification of ASSIGNMENT:
*Sep 17 14:20:59.784: DHCPD: address 192.168.152.1 mask 255.255.255.0
*Sep 17 14:20:59.784: DHCPD: htype 1 chaddr 000e.3525.f394
*Sep 17 14:20:59.784: DHCPD: table id 1 = vrf miovr
*Sep 17 14:20:59.784: DHCPD: lease time remaining (secs) = 86400

Note conclusive

A meno che non usiate il BGP, l'uso dell'RD in una configurazione VRF-lite e' secondario, basta il comando "ip vrf NOME" senza rd specificato per creare una nuova tabella di routing. L'RD è un identificativo di 8 byte che viene agganciato agli indirizzi IP creando un indirizzo VPN-IPv4 univoco all'interno della rete, nella forma RD:IP. Il BGP con il supporto alle estensioni multiprotocollo,RFC2858, viene utilizzato in configurazioni MPLS/VPN.

E' il protocollo BGP che puo' trasportare indirizzi RD:IP non IP, e' per questo che si usa nelle reti MPLS. BGP infatti e' multiprotocollo. Inoltre puo' trasportare altri dati come le communities, nel nostro caso eventuali valori route-target dove si indica, con i comandi "import/export" quali circuiti di una VPN possono vederne altri e cosi' via.

La configurazione di una rete VRF-lite con l'uso dei protocolli di routing in realta' e' cosa complessa. E' infatti importante conoscere a priori il funzionamento dettagliato dei vari algoritmi di routing, senza VRF, per poi passare al loro utilizzo in ambito VRF-lite.

Infine bisogna scegliere con attenzione l'immagine IOS dei router Cisco in uso, in quanto si corre il serio pericolo di non avere attive tutte le features necessarie (in particolare i protocolli di routing).

Versione 0.9 22 Agosto 2007

Copyright 2007 Gianrico Fichera

E' gradita la segnalazione di eventuali errori o imprecisioni. Scrivere a gianrico@gianrico.com.

Questo documento si puo' distribuire liberamente. Ogni uso differente dalla diffusione gratuita per uso didattico e' espressamente vietato senza autorizzazione espressa.

Il materiale di questo documento non e' sponsorizzato o sottoscritto da Cisco Systems, Inc. Cisco® e' un trademark di Cisco Systems, Inc. negli Stati Uniti e in altri stati. L'autore di queste pagine non si assume nessuna responsabilita' e non da nessuna garanzia riguardante l'accuratezza e la completezza delle informazioni presenti nonche' da conseguenze sull'uso delle informazioni presenti. Il sito web ufficiale della Cisco e' <http://www.cisco.com>. Nel caso si volesse utilizzare il contenuto di questo documento nella forma in cui e' presentato rivolgersi all'autore scrivendo a gianrico.fichera@itesys.it.