VPN IPsec multipoint dinamiche (tramite GRE/NHRP multipoint per scalare le VPN IPsec)

Sommario

Introduzione

In questo documento vengono descritte le VPN IPsec (DMVPN) dinamiche multipoint e le ragioni per cui un'azienda potrebbe voler progettare o migrare la rete per utilizzare questa nuova soluzione VPN IPsec nel software Cisco IOS®.

Premesse

Le aziende potrebbero dover interconnettere molti siti a un sito principale, e forse anche tra loro, attraverso Internet crittografando il traffico per proteggerlo. Ad esempio, una serie di punti vendita al dettaglio che devono connettersi alla sede centrale dell'azienda per effettuare l'inventario e l'ordinazione potrebbe anche dover connettersi ad altri punti vendita all'interno dell'azienda per verificare la disponibilità dei prodotti. In passato, l'unico modo per effettuare la connessione era usare una rete di layer 2, come ISDN o Frame Relay per interconnettere tutto. La configurazione e il pagamento di questi collegamenti cablati per il traffico IP interno possono essere dispendiosi in termini di tempo e denaro. Se tutti i siti (compreso quello principale) hanno già un accesso a Internet relativamente economico, questo accesso può essere utilizzato anche per le comunicazioni IP interne tra i negozi e la sede centrale, utilizzando i tunnel IPsec per garantire la privacy e l'integrità dei dati.

Per consentire alle aziende di creare reti IPsec di grandi dimensioni che interconnettono i propri siti tramite Internet, è necessario poter scalare la rete IPsec. IPsec cripta il traffico tra due endpoint (peer) e la crittografia viene eseguita dai due endpoint utilizzando un "segreto" condiviso. Poiché questo segreto viene condiviso solo tra questi due endpoint, le reti crittografate sono intrinsecamente una raccolta di collegamenti point-to-point. Per questo motivo, IPsec è intrinsecamente una rete tunnel point-to-point. Il metodo più semplice per scalare una rete point-to-point di grandi dimensioni è organizzarla in una rete hub-and-spoke o in una rete a maglia completa (parziale). Nella maggior parte delle reti, la maggior parte del traffico IP è tra gli spoke e l'hub, e molto poco è tra gli spoke, quindi la progettazione hub e spoke è spesso la scelta migliore. Questo design si abbina anche alle vecchie reti Frame Relay, poichè era proibitivo pagare i collegamenti tra tutti i siti in queste reti.

Quando si utilizza Internet come interconnessione tra hub e spoke, gli spoke hanno anche accesso diretto l'uno all'altro senza costi aggiuntivi, ma è stato molto difficile, se non impossibile, configurare e/o gestire una rete a maglia completa (parziale). Le reti mesh complete o parziali sono spesso auspicabili perché si possono ottenere risparmi sui costi se il traffico spoke può passare direttamente attraverso e non attraverso l'hub. Il traffico Spoke-to-Spoke che attraversa l'hub utilizza le risorse dell'hub e può causare ritardi aggiuntivi, in particolare quando si utilizza la crittografia IPsec, in quanto l'hub deve decrittografare i pacchetti in arrivo dagli spoke di invio e quindi crittografare nuovamente il traffico per inviarlo allo spoke di ricezione. Un altro esempio in cui il traffico diretto "spoke" sarebbe utile è il caso in cui due spoke si trovano nella stessa città e il hub è attraverso il paese.

Con l'implementazione e l'aumento delle dimensioni delle reti hub e spoke IPsec, è diventato più desiderabile utilizzare tali reti per instradare i pacchetti IP nel modo più dinamico possibile. Nelle vecchie reti hub e spoke Frame Relay questo veniva fatto eseguendo un protocollo di routing dinamico come OSPF o EIGRP sui collegamenti Frame Relay. Ciò è stato utile per pubblicizzare in modo dinamico la raggiungibilità delle reti spoke e per supportare la ridondanza nella rete di routing IP. Se la rete perde un router hub, un router hub di backup potrebbe assumere automaticamente il controllo per mantenere la connettività di rete alle reti spoke.

Esiste un problema fondamentale con i tunnel IPsec e i protocolli di routing dinamico. I protocolli di routing dinamico si basano sull'utilizzo di pacchetti IP multicast o broadcast, ma IPsec non supporta la crittografia di pacchetti multicast o broadcast. Per risolvere questo problema, il metodo corrente è usare i tunnel GRE (Generic Routing Encapsulation) in combinazione con la crittografia IPsec.

I tunnel GRE supportano il trasporto di pacchetti IP multicast e broadcast all'altra estremità del tunnel GRE. Il pacchetto del tunnel GRE è un pacchetto IP unicast, quindi il pacchetto GRE può essere crittografato con IPsec. In questo scenario, il GRE esegue il tunneling e IPsec esegue la parte di crittografia del supporto della rete VPN. Quando si configurano i tunnel GRE, gli indirizzi

IP degli endpoint del tunnel (**origine** del **tunnel**, **destinazione** del **tunnel**) devono essere noti all'altro endpoint e instradabili su Internet. Ciò significa che l'hub e tutti i router spoke della rete devono avere indirizzi IP statici non privati.

Per le connessioni di sito di piccole dimensioni a Internet, in genere l'indirizzo IP esterno di un spoke viene modificato ogni volta che si connette a Internet, in quanto il provider di servizi Internet (ISP) fornisce dinamicamente l'indirizzo dell'interfaccia esterna (tramite DHCP, Dynamic Host Configuration Protocol) ogni volta che il spoke viene connesso (ADSL (Asymmetric Digital Subscriber Line) e servizi via cavo). Questa allocazione dinamica dell'"indirizzo esterno" del router consente all'ISP di sovrascrivere l'uso del proprio spazio di indirizzi Internet, in quanto non tutti gli utenti saranno online contemporaneamente. Può essere molto più costoso pagare il provider per allocare un indirizzo statico per il router spoke. L'esecuzione di un protocollo di routing dinamico su una VPN IPsec richiede l'uso di tunnel GRE, ma non consente di avere spoke con indirizzi IP allocati dinamicamente sulle relative interfacce fisiche esterne.

Le restrizioni di cui sopra e alcune altre sono riassunte nei quattro punti seguenti:

- IPsec utilizza un elenco di controllo di accesso (ACL) per definire i dati da crittografare. Pertanto, ogni volta che si aggiunge una nuova (sotto)rete dietro uno spoke o un hub, il cliente deve modificare l'ACL su entrambi i router hub e spoke. Se lo Storage Processor gestisce il router, il cliente deve avvisare lo Storage Processor per ottenere la modifica dell'ACL IPsec in modo che il nuovo traffico venga crittografato.
- Con le reti hub e spoke di grandi dimensioni, le dimensioni della configurazione sul router hub possono diventare molto grandi, al punto da essere inutilizzabili. Ad esempio, un router hub deve avere fino a 3900 linee di configurazione per supportare 300 router spoke. Le dimensioni sono tali da rendere difficile la visualizzazione della configurazione e l'individuazione della sezione della configurazione relativa a un problema corrente di cui è in corso il debug. Inoltre, questa configurazione potrebbe essere troppo grande per la NVRAM e dovrebbe essere memorizzata nella memoria flash.
- GRE + IPsec deve conoscere l'indirizzo peer dell'endpoint. Gli indirizzi IP degli spoke sono connessi direttamente a Internet tramite il proprio ISP e spesso sono impostati in modo che i loro indirizzi di interfaccia esterna non siano fissi. Gli indirizzi IP possono essere modificati ogni volta che il sito viene connesso tramite DHCP.
- Se gli spoke devono comunicare direttamente tra loro sulla VPN IPsec, la rete hub e spoke deve diventare una rete a maglia completa. Poiché non è ancora noto quali raggi dovranno comunicare direttamente tra loro, è necessaria una rete completa, anche se ogni raggio potrebbe non avere bisogno di parlare direttamente con ogni altro raggio. Inoltre, non è possibile configurare IPSec su un router spoke di piccole dimensioni in modo che abbia connettività diretta con tutti gli altri router spoke nella rete; pertanto, i router spoke potrebbero dover essere router più potenti.

La soluzione DMVPN

La soluzione DMVPN utilizza Multipoint GRE (mGRE) e Next Hop Resolution Protocol (NHRP), con IPsec e alcuni nuovi miglioramenti, per risolvere i problemi sopra descritti in modo scalabile.

Avvio crittografia IPSec automatica

Quando non si utilizza la soluzione DMVPN, il tunnel di crittografia IPsec non viene avviato fino a

quando il traffico di dati non richiede l'utilizzo di questo tunnel IPsec. Il completamento dell'avvio del tunnel IPsec può richiedere da 1 a 10 secondi e il traffico di dati viene interrotto durante questo periodo. Quando si usa il GRE con IPsec, la configurazione del tunnel GRE include già l'indirizzo del peer del tunnel GRE (**destinazione del tunnel**), che è anche l'indirizzo del peer IPsec. Entrambi gli indirizzi sono preconfigurati.

Se si utilizzano il Tunnel Endpoint Discovery (TED) e le mappe crittografiche dinamiche sul router dell'hub, è possibile evitare di preconfigurare gli indirizzi peer IPsec sull'hub. Tuttavia, per poter avviare la negoziazione ISAKMP, è necessario inviare e ricevere una sonda e una risposta TED. Questa operazione non è necessaria perché, quando si utilizza il GRE, gli indirizzi di origine e di destinazione del peer sono già noti. Sono in configurazione o risolti con NHRP (per tunnel GRE multipoint).

Con la soluzione DMVPN, IPsec viene attivato immediatamente per i tunnel GRE point-to-point e multipoint. Inoltre, non è necessario configurare alcun ACL crittografico, in quanto verranno derivati automaticamente dagli indirizzi di origine e di destinazione del tunnel GRE. I comandi seguenti vengono utilizzati per definire i parametri di crittografia IPSec. Si noti che non sono richiesti comandi **set peer ...** o **match address ...** perché queste informazioni derivano direttamente dai mapping del tunnel GRE o NHRP associati.

```
crypto ipsec profile
```

```
 set transform-set
```

Il comando seguente associa un'interfaccia tunnel al profilo IPsec.

```
interface tunnel
```

```
 ...
 tunnel protection ipsec profile
```

Creazione dinamica del tunnel per i collegamenti "Spoke-to-Hub"

Nessuna informazione GRE o IPsec su un spoke configurata sul router hub nella rete DMVPN. Il tunnel GRE del router spoke è configurato (tramite comandi NHRP) con informazioni sul router

hub. Quando il router spoke si avvia, avvia automaticamente il tunnel IPsec con il router hub come descritto sopra. Quindi utilizza NHRP per notificare al router hub l'indirizzo IP dell'interfaccia fisica corrente. Ciò è utile per tre motivi:

- Se l'indirizzo IP dell'interfaccia fisica del router spoke è assegnato dinamicamente (ad esempio con ADSL o CableModem), il router hub non può essere configurato con queste informazioni poiché ogni volta che il router spoke viene ricaricato otterrà un nuovo indirizzo IP dell'interfaccia fisica.
- La configurazione del router hub è abbreviata e semplificata perché non richiede informazioni GRE o IPsec sui router peer. Tutte queste informazioni vengono apprese dinamicamente tramite NHRP.
- Quando si aggiunge un nuovo router spoke alla rete DMVPN, non è necessario modificare la configurazione sull'hub o su uno dei router spoke correnti. Il nuovo router spoke è configurato con le informazioni dell'hub e, quando viene avviato, si registra in modo dinamico con il router hub. Il protocollo di routing dinamico propaga all'hub le informazioni di routing per lo spoke. L'hub propaga le nuove informazioni di instradamento agli altri spoke. e propaga le informazioni di routing dagli altri spoke a questo spoke.

Creazione dinamica del tunnel per il traffico "Spoke-to-Spoke"

Come accennato in precedenza, al momento in una rete mesh, tutti i tunnel IPsec (o IPsec+GRE) point-to-point devono essere configurati su tutti i router, anche se alcuni/la maggior parte di questi tunnel non sono in esecuzione o necessari in qualsiasi momento. Con la soluzione DMVPN, un router è l'hub e tutti gli altri router (spoke) sono configurati con tunnel per l'hub. I tunnel spoke-to-hub sono attivi continuamente e gli spoke non richiedono configurazione per tunnel diretti ad alcun altro spoke. Al contrario, quando un spoke desidera trasmettere un pacchetto a un altro spoke (come la subnet dietro un altro spoke), utilizza NHRP per determinare in modo dinamico l'indirizzo di destinazione richiesto del spoke di destinazione. Il router hub funge da server NHRP e gestisce questa richiesta per lo spoke di origine. I due spoke creano quindi dinamicamente un tunnel IPsec tra di essi (tramite l'interfaccia singola mGRE) e i dati possono essere trasferiti direttamente. Questo tunnel dinamico spoke-to-spoke verrà automaticamente disattivato dopo un periodo di inattività (configurabile).

Supporto dei protocolli di routing dinamico

La soluzione DMVPN si basa sui tunnel GRE che supportano il tunneling di pacchetti IP multicast/broadcast, quindi la soluzione DMVPN supporta anche i protocolli di routing dinamico in esecuzione sui tunnel IPsec+mGRE. In precedenza, NHRP richiedeva di configurare in modo esplicito il mapping broadcast/multicast per gli indirizzi IP di destinazione del tunnel per supportare il tunneling GRE dei pacchetti multicast e Broadcast IP. Ad esempio, sull'hub è necessaria la linea di configurazione **ip nhrp map <spoke-n-addr>** per ogni spoke. Con la soluzione DMVPN, gli indirizzi spoke non sono noti in anticipo, quindi questa configurazione non è possibile. Al contrario, NHRP può essere configurato per aggiungere automaticamente ogni spoke all'elenco di destinazione multicast sull'hub con il comando **ip nhrp map multicast dynamic**. Con questo comando, quando i router spoke registrano il proprio mapping NHRP unicast con il server NHRP (hub), NHRP crea anche un mapping broadcast/multicast per questo spoke. In questo modo non è più necessario che gli indirizzi siano noti in anticipo.

Cisco Express Forwarding Fast Switching per mGRE

Al momento, il traffico in un'interfaccia mGRE è a commutazione di contesto, il che determina prestazioni scadenti. La soluzione DMVPN aggiunge la commutazione Cisco Express Forwarding per il traffico GRE, consentendo di migliorare notevolmente le prestazioni. Non sono necessari comandi di configurazione per attivare questa funzione. Se la commutazione di inoltro Cisco Express è consentita sull'interfaccia del tunnel GRE e sulle interfacce fisiche in uscita/in entrata, i pacchetti del tunnel GRE multipoint saranno commutati in inoltro Cisco Express.

Uso del routing dinamico su VPN protette IPsec

In questa sezione viene descritto lo stato attuale (soluzione precedente a DMVPN). IPsec viene implementato sui router Cisco tramite un set di comandi che definiscono la crittografia e quindi un comando **crypto map <map-name>** applicato sull'interfaccia esterna del router. A causa di questa progettazione e del fatto che non esiste attualmente uno standard per utilizzare IPsec per crittografare i pacchetti IP multicast/broadcast, i pacchetti del protocollo di routing IP non possono essere "inoltrati" tramite il tunnel IPsec e le modifiche di routing non possono essere propagate dinamicamente sull'altro lato del tunnel IPsec.

Nota: tutti i protocolli di routing dinamico, ad eccezione di BGP, utilizzano pacchetti IP broadcast o multicast. Per risolvere il problema, vengono utilizzati i tunnel GRE in combinazione con IPsec.

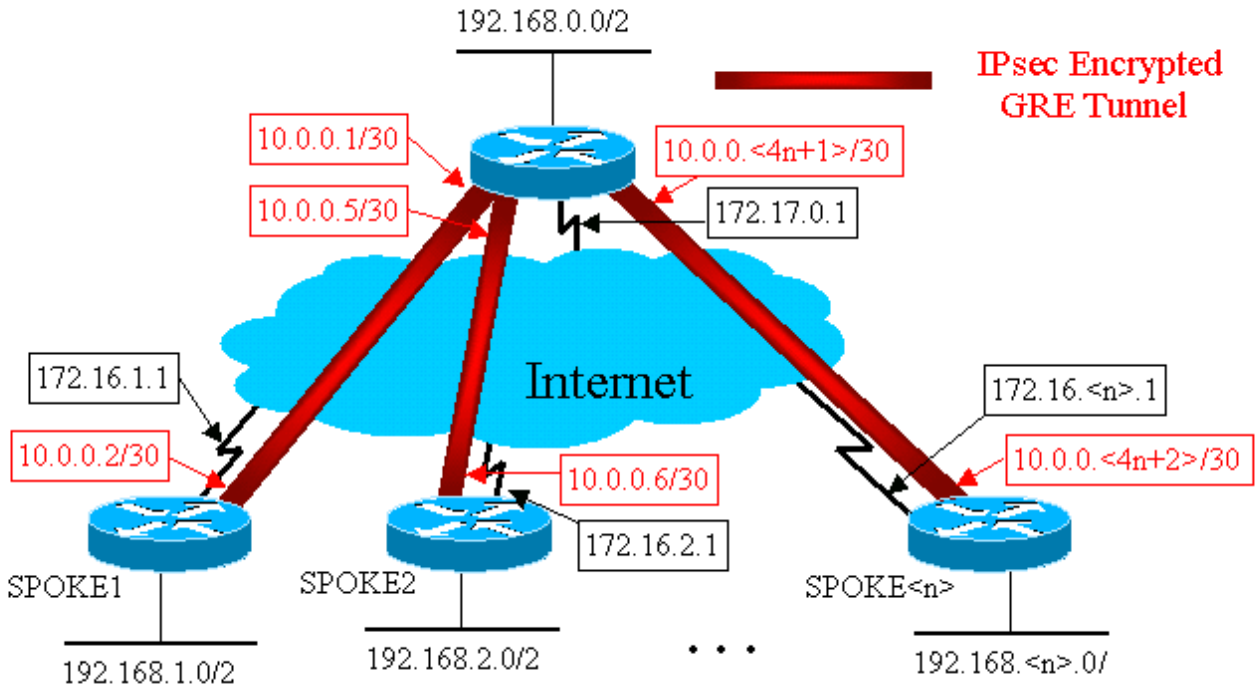
I tunnel GRE vengono implementati sui router Cisco tramite un'interfaccia di tunnel virtuale (**interface tunnel<#>**). Il protocollo di tunneling GRE è progettato per gestire pacchetti IP multicast/broadcast in modo che un protocollo di routing dinamico possa essere "eseguito" su un tunnel GRE. I pacchetti del tunnel GRE sono pacchetti IP unicast che incapsulano il pacchetto IP multicast/unicast originale. È quindi possibile utilizzare IPsec per crittografare il pacchetto del tunnel GRE. È possibile anche eseguire IPsec in modalità trasporto e salvare 20 byte, in quanto il GRE ha già incapsulato il pacchetto dati originale, quindi non è necessario usare IPsec per incapsulare il pacchetto GRE IP in un'altra intestazione IP.

Quando si esegue IPsec in modalità trasporto, esiste una restrizione secondo cui gli indirizzi di origine e di destinazione IP del pacchetto da crittografare devono corrispondere agli indirizzi peer IPsec (il router stesso). In questo caso, significa solo che l'endpoint del tunnel GRE e gli indirizzi peer IPsec devono essere uguali. Questo non è un problema in quanto gli stessi router sono entrambi gli endpoint del tunnel IPsec e GRE. Combinando i tunnel GRE con la crittografia IPsec, è possibile utilizzare un protocollo di routing IP dinamico per aggiornare le tabelle di routing su entrambe le estremità del tunnel crittografato. Le voci della tabella di routing IP per le reti apprese tramite il tunnel crittografato useranno l'altra estremità del tunnel (indirizzo IP dell'interfaccia del tunnel GRE) come hop IP successivo. Pertanto, se le reti cambiano su entrambi i lati del tunnel, l'altro lato verrà a conoscenza dinamicamente della modifica e la connettività continuerà senza alcuna modifica alla configurazione sui router.

Configurazione di base

Di seguito è riportata una configurazione standard IPsec+GRE point-to-point. In seguito, è disponibile una serie di esempi di configurazione in cui vengono aggiunte funzionalità specifiche della soluzione DMVPN in passaggi che mostrano le diverse funzionalità di DMVPN. Ogni esempio si basa sugli esempi precedenti per mostrare come utilizzare la soluzione DMVPN in progetti di rete sempre più complessi. Questa serie di esempi può essere utilizzata come modello per la migrazione di una VPN IPsec+GRE corrente a una VPN DMVPN. È possibile interrompere la "migrazione" in qualsiasi momento se l'esempio di configurazione soddisfa i requisiti di progettazione della rete.

IPsec + hub e spoke GRE (n = 1,2,3,...)



Router hub

```
version 12.3
 !
 hostname Hub
 !
 crypto isakmp policy 1
  authentication pre-share
 crypto isakmp key cisco47 address 0.0.0.0
 !
 crypto ipsec transform-set trans2 esp-des esp-md5-hmac
  mode transport
 !
 crypto map vpnmap1 local-address Ethernet0
 crypto map vpnmap1 10 ipsec-isakmp
  set peer 172.16.1.1
  set transform-set trans2
  match address 101
 crypto map vpnmap1 20 ipsec-isakmp
  set peer 172.16.2.1
  set transform-set trans2
  match address 102
 . . .
 crypto map vpnmap1 <10*n> ipsec-isakmp
  set peer 172.16.




 interface Tunnel1
  bandwidth 1000
  ip address 10.0.0.1 255.255.255.252
  ip mtu 1400
  delay 1000
```

```
 tunnel source Ethernet0
 tunnel destination 172.16.1.1
!
interface Tunnel2
 bandwidth 1000
 ip address 10.0.0.5 255.255.255.252
 ip mtu 1400
 delay 1000
 tunnel source Ethernet0
 tunnel destination 172.16.2.1
!
. . .
!
interface Tunnel




!
interface Ethernet0
 ip address 172.17.0.1 255.255.255.0
 crypto map vpnmap1
!
interface Ethernet1
 ip address 192.168.0.1 255.255.255.0
!
router eigrp 1
 network 10.0.0.0 0.0.0.255
 network 192.168.0.0 0.0.0.255
 no auto-summary
!
access-list 101 permit gre host 172.17.0.1 host
172.16.1.1
access-list 102 permit gre host 172.17.0.1 host
172.16.2.1
...
access-list
```

Router Spoke1

```
version 12.3
!
hostname Spoke1
!
crypto isakmp policy 1
 authentication pre-share
crypto isakmp key cisco47 address 0.0.0.0
!
crypto ipsec transform-set trans2 esp-des esp-md5-hmac
 mode transport
!
crypto map vpnmap1 local-address Ethernet0
crypto map vpnmap1 10 ipsec-isakmp
 set peer 172.17.0.1
 set transform-set trans2
 match address 101
!
```

```
 interface Tunnel0
  bandwidth 1000
  ip address 10.0.0.2 255.255.255.252
  ip mtu 1400
  delay 1000
  tunnel source Ethernet0
  tunnel destination 172.17.0.1
 !
 interface Ethernet0
  ip address 172.16.1.1 255.255.255.252
  crypto map vpnmap1
 !
 interface Ethernet1
  ip address 192.168.1.1 255.255.255.0
 !
 router eigrp 1
  network 10.0.0.0 0.0.0.255
  network 192.168.1.0 0.0.0.255
  no auto-summary
 !
 access-list 101 permit gre host 172.16.1.1 host
172.17.0.1
```

Router Spoke2

```
version 12.3
 !
 hostname Spoke2
 !
 crypto isakmp policy 1
  authentication pre-share
 crypto isakmp key cisco47 address 0.0.0.0
 !
 crypto ipsec transform-set trans2 esp-des esp-md5-hmac
  mode transport
 !
 crypto map vpnmap1 local-address Ethernet0
 crypto map vpnmap1 10 ipsec-isakmp
  set peer 172.17.0.1
  set transform-set trans2
  match address 101
 !
 interface Tunnel0
  bandwidth 1000
  ip address 10.0.0.6 255.255.255.252
  ip mtu 1400
  delay 1000
  tunnel source Ethernet0
  tunnel destination 172.17.0.1
 !
 interface Ethernet0
  ip address 172.16.2.1 255.255.255.252
  crypto map vpnmap1
 !
 interface Ethernet1
  ip address 192.168.2.1 255.255.255.0
 !
 router eigrp 1
  network 10.0.0.0 0.0.0.255
  network 192.168.2.0 0.0.0.255
  no auto-summary
 !
 access-list 101 permit gre host 172.16.2.1 host
```

172.17.0.1

Router Spoke<n>

```
version 12.3
 !
 hostname Spoke<n>
 !
 crypto isakmp policy 1
  authentication pre-share
 crypto isakmp key cisco47 address 0.0.0.0
 !
 crypto ipsec transform-set trans2 esp-des esp-md5-hmac
  mode transport
 !
 crypto map vpnmap1 local-address Ethernet0
 crypto map vpnmap1 10 ipsec-isakmp
  set peer 172.17.0.1
  set transform-set trans2
  match address 101
 !
 interface Tunnel0
  bandwidth 1000
  ip address 10.0.0.<4n-2> 255.255.255.252
  ip mtu 1400
  delay 1000
  tunnel source Ethernet0
  tunnel destination 172.17.0.1
 !
 interface Ethernet0
  ip address 172.16.<n>.1 255.255.255.252
  crypto map vpnmap1
 !
 interface Ethernet1
  ip address 192.168.<n>.1 255.255.255.0
 !
 router eigrp 1
  network 10.0.0.0 0.0.0.255
  network 192.168.
```

Nella configurazione precedente, gli ACL vengono usati per definire il traffico che verrà crittografato. Sui router hub e spoke, questo ACL deve corrispondere solo ai pacchetti IP del tunnel GRE. Indipendentemente dal cambiamento delle reti a entrambe le estremità, i pacchetti del tunnel GRE IP non cambieranno, quindi questo ACL non deve essere modificato.

Nota: quando si usa il software Cisco IOS con versioni precedenti alla 12.2(13)T, è necessario applicare il comando di configurazione **crypto map vpnmap1** sia all'interfaccia del tunnel GRE (Tunnel<x>) sia all'interfaccia fisica (Ethernet0). Con Cisco IOS versione 12.2(13)T e successive, il comando di configurazione **crypto map vpnmap1** viene applicato solo all'interfaccia fisica (Ethernet0).

Esempi di tabelle di routing sui router hub e spoke

Tabella di routing sul router hub

```
172.17.0.0/24 is subnetted, 1 subnets
 C       172.17.0.0 is directly connected, Ethernet0
      10.0.0.0/30 is subnetted, <n> subnets
 C       10.0.0.0 is directly connected, Tunnel1
 C       10.0.0.4 is directly connected, Tunnel2
 ...
 C       10.0.0.<4n-4> is directly connected, Tunnel<n>
 C    192.168.0.0/24 is directly connected, Ethernet1
 D    192.168.1.0/24 [90/2841600] via 10.0.0.2,
18:28:19, Tunnel1
 D    192.168.2.0/24 [90/2841600] via 10.0.0.6, 2d05h,
Tunnel2
 ...
 D    192.168.<n>.0/24 [90/2841600] via 10.0.0.<4n-2>,
2d05h, Tunnel<n>
```

Tabella di routing su router Spoke1

```
172.16.0.0/24 is subnetted, 1 subnets
 C       172.16.1.0 is directly connected, Ethernet0
      10.0.0.0/30 is subnetted, <n> subnets
 C       10.0.0.0 is directly connected, Tunnel1
 D       10.0.0.4 [90/2841600] via 10.0.0.1, 23:00:58,
Tunnel0
 ...
 D      10.0.0.<4n-4> [90/2841600] via 10.0.0.1,
23:00:58, Tunnel0
 D    192.168.0.0/24 [90/2841600] via 10.0.0.1,
23:00:58, Tunnel0
 C    192.168.1.0/24 is directly connected, Loopback0
 D    192.168.2.0/24 [90/3097600] via 10.0.0.1,
23:00:58, Tunnel0
 ...
 D    192.168.<n>.0/24 [90/3097600] via 10.0.0.1,
23:00:58, Tunnel0
```

Tabella di routing su router Spoke<n>

```
172.16.0.0/24 is subnetted, 1 subnets
 C       172.16.<n>.0 is directly connected, Ethernet0
      10.0.0.0/30 is subnetted, <n> subnets
 D       10.0.0.0 [90/2841600] via 10.0.0.1, 22:01:21,
Tunnel0
 D       10.0.0.4 [90/2841600] via 10.0.0.1, 22:01:21,
Tunnel0
 ...
 C        10.0.0.<4n-4> is directly connected, Tunnel0
 D    192.168.0.0/24 [90/2841600] via 10.0.0.1,
22:01:21, Tunnel0
 D    192.168.1.0/24 [90/3097600] via 10.0.0.1,
22:01:21, Tunnel0
 D    192.168.2.0/24 [90/3097600] via 10.0.0.1,
22:01:21, Tunnel0
 ...
 C    192.168.<n>.0/24 is directly connected, Ethernet0
```

Si tratta di una configurazione operativa di base, utilizzata come punto di partenza per il confronto

con le configurazioni più complesse possibili utilizzando la soluzione DMVPN. La prima modifica ridurrà le dimensioni della configurazione sul router hub. Questa operazione non è importante con un numero ridotto di router di spoke, ma diventa critica quando sono presenti più di 50-100 router di spoke.

Riduzione delle dimensioni della configurazione del router hub

Nell'esempio seguente, la configurazione viene modificata almeno sul router hub da più interfacce del tunnel GRE point-to-point a una singola interfaccia del tunnel GRE multipoint. Questo è un primo passo verso la soluzione DMVPN.

Sul router hub è presente un blocco univoco di linee di configurazione per definire le caratteristiche della mappa crittografica per ciascun router spoke. Questa parte della configurazione definisce l'ACL crittografico e l'interfaccia del tunnel GRE per il router spoke. Queste caratteristiche sono per lo più le stesse per tutti gli spoke, ad eccezione degli indirizzi IP (**set peer ...**, **destinazione tunnel ...**).

Osservando la configurazione precedente sul router hub, si osserverà che ci sono almeno 13 linee di configurazione per router spoke; quattro per la mappa crittografica, uno per l'ACL crittografico e otto per l'interfaccia del tunnel GRE. Il numero totale di linee di configurazione, se erano presenti 300 router a raggi, è 3900 linee. Inoltre, sono necessarie 300 (/30) subnet per indirizzare ciascun collegamento del tunnel. Una configurazione di queste dimensioni è molto difficile da gestire e ancora più difficile da gestire quando si risolvono i problemi della rete VPN. Per ridurre questo valore, è possibile utilizzare le mappe crittografiche dinamiche, che ridurrebbero il valore sopra indicato di 1200 linee, lasciando 2700 linee in una rete a 300 raggi.

Nota: quando si utilizzano mappe crittografiche dinamiche, il tunnel di crittografia IPsec deve essere avviato dal router spoke. È inoltre possibile utilizzare il comando **ip unnumber <interface>** per ridurre il numero di subnet necessarie per i tunnel GRE, ma questa operazione potrebbe rendere più difficile la risoluzione dei problemi in un secondo momento.

Con la soluzione DMVPN, è possibile configurare una singola interfaccia del tunnel GRE multipoint e un singolo profilo IPsec sul router hub per gestire tutti i router spoke. In questo modo, le dimensioni della configurazione sul router hub rimangono costanti, a prescindere dal numero di router spoke aggiunti alla rete VPN.

La soluzione DMVPN introduce i seguenti nuovi comandi:

```
crypto ipsec profile
```

Il comando **crypto ipsec profile <name>** viene usato come mappa crittografica dinamica e è progettato specificamente per le interfacce tunnel. Questo comando è usato per definire i parametri per la crittografia IPsec sui tunnel VPN spoke-to-hub e spoke-to-spoke. L'unico parametro richiesto dal profilo è il set di trasformazioni. L'indirizzo peer IPsec e la clausola **match address ...** per il proxy IPsec vengono derivati automaticamente dai mapping NHRP per il tunnel GRE.

Il comando **tunnel protection ipsec profile <name>** è configurato nell'interfaccia del tunnel GRE e viene usato per associare l'interfaccia del tunnel GRE al profilo IPsec. Inoltre, il comando **tunnel protection ipsec profile <name>** può essere usato anche con un tunnel GRE point-to-point. In questo caso, le informazioni sul peer IPsec e sul proxy verranno derivate dalla configurazione di **origine** del **tunnel** e **destinazione** del **tunnel**. Ciò semplifica la configurazione in quanto il peer IPsec e gli ACL crittografici non sono più necessari.

Nota: il comando **tunnel protection ...** specifica che la crittografia IPsec verrà eseguita dopo l'aggiunta dell'incapsulamento GRE al pacchetto.

I primi due nuovi comandi sono simili alla configurazione di una mappa crittografica e all'assegnazione della mappa crittografica a un'interfaccia utilizzando il comando **crypto map<name>**. La grande differenza è che, con i nuovi comandi, non è necessario specificare l'indirizzo peer IPsec o un ACL che corrisponda ai pacchetti da crittografare. Questi parametri vengono determinati automaticamente dai mapping NHRP per l'interfaccia del tunnel GRE.

Nota: quando si usa il comando **tunnel protection ...** sull'interfaccia del tunnel, un comando **crypto map ...** non è configurato sull'interfaccia fisica in uscita.

L'ultimo nuovo comando, **ip nhrp map multicast dynamic**, consente a NHRP di aggiungere automaticamente router spoke ai mapping NHRP multicast quando questi router spoke avviano il tunnel GRE+IPsec e registrano i propri mapping NHRP unicast. Questa operazione è necessaria per consentire il funzionamento dei protocolli di routing dinamico sui tunnel GRE+IPsec tra l'hub e gli spoke. Se questo comando non è disponibile, il router hub deve disporre di una riga di configurazione separata per un mapping multicast a ogni spoke.

Nota: con questa configurazione, i router spoke devono avviare la connessione al tunnel GRE+IPsec, poiché il router hub non è configurato con alcuna informazione sugli spoke. Tuttavia, non si tratta di un problema perché con DMVPN il tunnel GRE+IPsec viene avviato automaticamente all'avvio del router spoke e rimane sempre attivo.

Nota: l'esempio che segue mostra le interfacce del tunnel GRE point-to-point sui router spoke e le linee della configurazione NHRP aggiunta sui router hub e spoke per supportare il tunnel GRE sul router hub. Le modifiche alla configurazione sono le seguenti.

Router hub (vecchio)

```
crypto map vpnmap1 10 IPsec-isakmp
  set peer 172.16.1.1
  set transform-set trans2
  match address 101
 crypto map vpnmap1 20 IPsec-isakmp
  set peer 172.16.2.1
  set transform-set trans2
  match address 102
 . . .
 crypto map vpnmap1 <10n> IPsec-isakmp
  set peer 172.16.




 !
```

```
 interface Ethernet0
  ip address 172.17.0.1 255.255.255.0
  crypto map vpnmap1
 !
 access-list 101 permit gre host 172.17.0.1 host
172.16.1.1
 access-list 102 permit gre host 172.17.0.1 host
172.16.2.1
 . . .
 access-list
```

Router hub (nuovo)

```
crypto ipsec profile vpnprof
  set transform-set trans2
 !
 interface Tunnel0
  bandwidth 1000
  ip address 10.0.0.1 255.255.255.0
  ip mtu 1400
  ip nhrp authentication test
  ip nhrp map multicast dynamic
  ip nhrp network-id 100000
  ip nhrp holdtime 600
  no ip split-horizon eigrp 1
  delay 1000
  tunnel source Ethernet0
  tunnel mode gre multipoint
  tunnel key 100000
  tunnel protection ipsec profile vpnprof
 !
 interface Ethernet0
  ip address 172.17.0.1 255.255.255.0
```

Router Spoke<n> (vecchio)

```
crypto map vpnmap1 10 IPsec-isakmp
  set peer 172.17.0.1
  set transform-set trans2
  match address 101
 !
 interface Tunnel0
  bandwidth 1000
  ip address 10.0.0.<4n-2> 255.255.255.252
  ip mtu 1400
  delay 1000
  tunnel source Ethernet0
  tunnel destination 172.17.0.1
 !
interface Ethernet0
  ip address 172.16.<n>.1 255.255.255.252
  crypto map vpnmap1
!
 . . .
 !
```

```
 access-list 101 permit gre host 172.16.<n>.1 host
172.17.0.1
 !
```

Spoke<n> Router (nuovo)

```
crypto map vpnmap1 10 IPsec-isakmp
  set peer 172.17.0.1
  set transform-set trans2
  match address 101
 !
 interface Tunnel0
  bandwidth 1000
  ip address 10.0.0.




  delay 1000
  tunnel source Ethernet0
  tunnel destination 172.17.0.1
  tunnel key 100000
 !
interface Ethernet0
  ip address 172.16.<n>.1 255.255.255.252
  crypto map vpnmap1
!
 . . .
 !
 access-list 101 permit gre host 172.16.<n>.1 host
172.17.0.1
 !
```

Sui router spoke, la subnet mask è stata modificata e sono stati aggiunti comandi NHRP nell'interfaccia del tunnel. I comandi NHRP sono necessari perché il router hub sta ora utilizzando NHRP per mappare l'indirizzo IP dell'interfaccia del tunnel spoke all'indirizzo IP dell'interfaccia fisica spoke.

```
ip address 10.0.0.
```

```
      ip mtu 1400
      ip nhrp authentication test
      ip nhrp map 10.0.0.1 172.17.0.1
      ip nhrp network-id 100000
      ip nhrp holdtime 300
      ip nhrp nhs 10.0.0.1
      ...
      tunnel key 100000
```

La subnet è ora /24 anziché /30, pertanto tutti i nodi si trovano nella stessa subnet, anziché in subnet diverse. Gli spoke inviano ancora traffico spoke tramite l'hub, in quanto utilizzano un'interfaccia del tunnel GRE point-to-point. I comandi **ip nhrp authentication ...**, **ip nhrp network-id ...** e **tunnel key ...** vengono usati per mappare i pacchetti del tunnel e i pacchetti NHRP

all'interfaccia del tunnel GRE multipoint e alla rete NHRP corretti quando vengono ricevuti sull'hub. I comandi **ip nhrp map ...** e **ip nhrp nhs** ... vengono utilizzati da NHRP sullo spoke per annunciare il mapping NHRP degli spoke (10.0.0.<n+1> —> 172.16.<n>.1) all'hub. L'indirizzo 10.0.0.<n+1> viene recuperato dal comando **ip address ...** sull'interfaccia del tunnel e l'indirizzo 172.16.<n>.1 viene recuperato dal comando **tunnel destination ...** sull'interfaccia del tunnel.

Nel caso in cui vi siano 300 router spoke, questa modifica ridurrebbe il numero di linee di configurazione sull'hub da 3900 linee a 16 linee (una riduzione di 3884 linee). La configurazione su ciascun router spoke aumenterebbe di 6 linee.

Supporto di indirizzi dinamici su raggi

Su un router Cisco, ogni peer IPsec deve essere configurato con l'indirizzo IP dell'altro peer IPsec prima di poter visualizzare il tunnel IPsec. Questa operazione ha un problema se un router spoke ha un indirizzo dinamico sulla sua interfaccia fisica, cosa comune per i router connessi tramite DSL o collegamenti via cavo.

Il protocollo TED consente a un peer IPsec di trovare un altro peer IPsec inviando un pacchetto ISAKMP (Internet Security Association and Key Management Protocol) speciale all'indirizzo di destinazione IP del pacchetto dati originale da crittografare. Si presume che il pacchetto attraversi la rete in ingresso seguendo lo stesso percorso del pacchetto del tunnel IPsec. Questo pacchetto verrà raccolto dall'altro peer IPsec finale, che risponderà al primo peer. I due router negozieranno quindi le associazioni di sicurezza ISAKMP e IPsec e avvieranno il tunnel IPsec. Questa operazione funziona solo se i pacchetti di dati da crittografare hanno indirizzi IP instradabili.

Il formato TED può essere usato in combinazione con i tunnel GRE come configurato nella sezione precedente. Questo è stato testato e funziona, anche se nelle versioni precedenti del software Cisco IOS il TED ha forzato la crittografia di tutto il traffico IP tra i due peer IPsec, non solo dei pacchetti del tunnel GRE. La soluzione DMVPN fornisce queste e altre funzionalità senza che gli host debbano utilizzare indirizzi IP instradabili su Internet e senza dover inviare pacchetti di richiesta e risposta. Con una lieve modifica, la configurazione dell'ultima sezione può essere utilizzata per supportare router spoke con indirizzi IP dinamici sulle loro interfacce fisiche esterne.

Router hub (nessuna modifica)

```
crypto ipsec profile vpnprof
  set transform-set trans2
 !
 interface Tunnel0
  bandwidth 1000
  ip address 10.0.0.1 255.255.255.0
  ip mtu 1400
  ip nhrp authentication test
  ip nhrp map multicast dynamic
  ip nhrp network-id 100000
  ip nhrp holdtime 600
  no ip split-horizon eigrp 1
  delay 1000
  tunnel source Ethernet0
  tunnel mode gre multipoint
  tunnel key 100000
  tunnel protection ipsec profile vpnprof
 !
 interface Ethernet0
```

```
  ip address 172.17.0.1 255.255.255.0
```

Router Spoke<n> (vecchio)

```
crypto map vpnmap1 10 IPsec-isakmp
  set peer 172.17.0.1
  set transform-set trans2
  match address 101
 !
 ...
 !
 access-list 101 permit gre host 172.16.
```

Spoke<n> Router (nuovo)

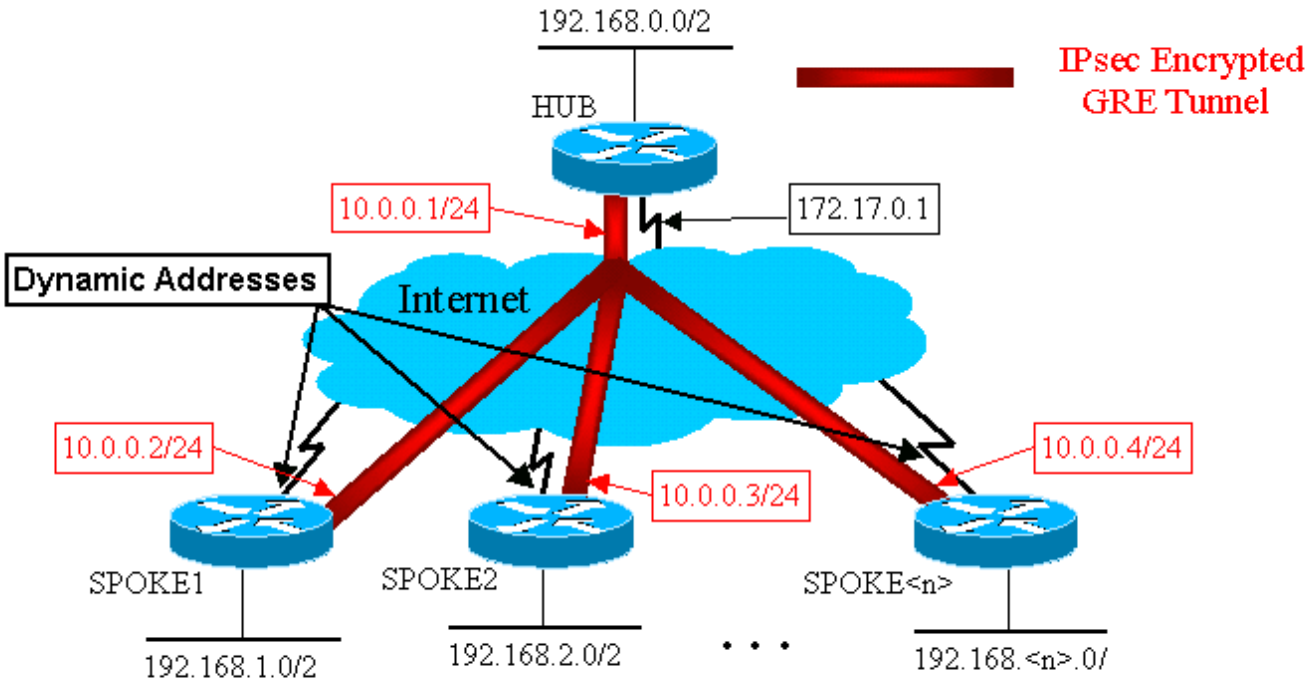
```
crypto map vpnmap1 10 IPsec-isakmp
  set peer 172.17.0.1
  set transform-set trans2
  set security-association level per-host
  match address 101
 !
 ...
 !
 access-list 101 permit gre any host 172.17.0.1
```

La funzionalità utilizzata nella nuova configurazione spoke è la seguente.

- Quando l'interfaccia del tunnel GRE è attiva, inizierà a inviare i pacchetti di registrazione NHRP al router hub. Questi pacchetti di registrazione NHRP attiveranno l'avvio di IPSec. Sul router spoke, vengono configurati i comandi set peer <indirizzo-peer> e **match ip access-list<ACL>**. Nell'ACL, il protocollo GRE è specificato come protocollo, tutti per l'origine e l'indirizzo IP dell'hub per la destinazione.**Nota:** è importante notare che nell'ACL viene usato any (qualsiasi) come origine, e questo deve avvenire perché l'indirizzo IP del router spoke è dinamico e, quindi, non è noto prima che l'interfaccia fisica sia attiva. È possibile usare una subnet IP per l'origine nell'ACL se l'indirizzo dell'interfaccia spoke dinamica è limitato a un indirizzo all'interno di quella subnet.
- Il comando **set security-association level per-host** viene usato in modo che l'origine IP nel proxy IPsec di spoke sia solo l'indirizzo dell'interfaccia fisica corrente di spoke (/32), piuttosto che il valore "any" (qualsiasi) restituito dall'ACL. Se si usasse "qualsiasi" dall'ACL come origine nel proxy IPsec, si impedirebbe a qualsiasi router spoke di configurare anche un tunnel IPsec+GRE con questo hub. Infatti, il proxy IPsec risultante sull'hub equivale a **consentire** all'**host gre 172.17.0.1 qualsiasi**. Questo significherebbe che tutti i pacchetti del tunnel GRE destinati a qualsiasi spoke verrebbero crittografati e inviati al primo spoke che stabilisse un tunnel con l'hub, poiché il suo proxy IPsec corrisponde ai pacchetti GRE per ogni spoke.
- Dopo aver configurato il tunnel IPsec, un pacchetto di registrazione NHRP viene inviato dal router spoke al server NHS (Next Hop Server) configurato. NHS è il router hub di questa rete hub e spoke. Il pacchetto di registrazione NHRP fornisce le informazioni necessarie al router

hub per creare un mapping NHRP per questo router spoke. Con questa mappatura, il router hub può quindi inoltrare pacchetti di dati IP unicast a questo router spoke sul tunnel GRE+IPsec. Inoltre, l'hub aggiunge il router spoke al proprio elenco di mapping multicast NHRP. L'hub inizierà quindi a inviare pacchetti multicast di routing IP dinamico allo spoke (se è configurato un protocollo di routing dinamico). Lo spoke diventerà quindi un protocollo di routing adiacente all'hub e scambierà gli aggiornamenti di routing.

IPsec + hub e spoke GRE



Router hub

```
version 12.3
 !
 hostname Hub
 !
 crypto isakmp policy 1
  authentication pre-share
 crypto isakmp key cisco47 address 0.0.0.0
 !
 crypto ipsec transform-set trans2 esp-des esp-md5-hmac
  mode transport
 !
 crypto ipsec profile vpnprof
  set transform-set trans2
 !
 interface Tunnel0
  bandwidth 1000
  ip address 10.0.0.1 255.255.255.0
  ip mtu 1400
  ip nhrp authentication test
  ip nhrp map multicast dynamic
  ip nhrp network-id 100000
  ip nhrp holdtime 600
```

```
 no ip split-horizon eigrp 1
 delay 1000
 tunnel source Ethernet0
 tunnel mode gre multipoint
 tunnel key 100000
 tunnel protection ipsec profile vpnprof
!
interface Ethernet0
 ip address 172.17.0.1 255.255.255.0
!
interface Ethernet1
 ip address 192.168.0.1 255.255.255.0
!
router eigrp 1
 network 10.0.0.0 0.0.0.255
 network 192.168.0.0 0.0.0.255
 no auto-summary
!
```

Nella configurazione dell'hub sopra riportata gli indirizzi IP dei router spoke non sono configurati. L'interfaccia fisica esterna dello spoke e la mappatura agli indirizzi IP dell'interfaccia del tunnel dello spoke vengono apprese dinamicamente dall'hub tramite NHRP. Questo consente l'assegnazione dinamica dell'indirizzo IP dell'interfaccia fisica esterna del spoke.

Router Spoke1

```
version 12.3
!
hostname Spoke1
!
crypto isakmp policy 1
 authentication pre-share
crypto isakmp key cisco47 address 0.0.0.0 0.0.0.0
!
crypto ipsec transform-set trans2 esp-des esp-md5-hmac
 mode transport
!
crypto map vpnmap1 local-address Ethernet0
crypto map vpnmap1 10 IPsec-isakmp
 set peer 172.17.0.1
 set security-association level per-host
 set transform-set trans2
 match address 101
!
interface Tunnel0
 bandwidth 1000
 ip address 10.0.0.2 255.255.255.0
 ip mtu 1400
 ip nhrp authentication test
 ip nhrp map 10.0.0.1 172.17.0.1
 ip nhrp network-id 100000
 ip nhrp holdtime 300
 ip nhrp nhs 10.0.0.1
 delay 1000
 tunnel source Ethernet0
 tunnel destination 172.17.0.1
 tunnel key 100000
!
interface Ethernet0
 ip address dhcp hostname Spoke1
```

```
 crypto map vpnmap1
!
interface Ethernet1
 ip address 192.168.1.1 255.255.255.0
!
router eigrp 1
 network 10.0.0.0 0.0.0.255
 network 192.168.1.0 0.0.0.255
 no auto-summary
!
access-list 101 permit gre 172.16.1.0 0.0.0.255 host
172.17.0.1
```


```
version 12.3
!
hostname Spoke2
!
crypto isakmp policy 1
 authentication pre-share
crypto isakmp key cisco47 address 0.0.0.0 0.0.0.0
!
crypto ipsec transform-set trans2 esp-des esp-md5-hmac
 mode transport
!
crypto map vpnmap1 local-address Ethernet0
crypto map vpnmap1 10 IPsec-isakmp
 set peer 172.17.0.1
 set security-association level per-host
 set transform-set trans2
 match address 101
!
interface Tunnel0
 bandwidth 1000
 ip address 10.0.0.3 255.255.255.0
 ip mtu 1400
 ip nhrp authentication test
 ip nhrp map 10.0.0.1 172.17.0.1
 ip nhrp network-id 100000
 ip nhrp holdtime 300
 ip nhrp nhs 10.0.0.1
 delay 1000
 tunnel source Ethernet0
 tunnel destination 172.17.0.1
 tunnel key 100000
!
interface Ethernet0
 ip address dhcp hostname Spoke2
 crypto map vpnmap1
!
interface Ethernet1
 ip address 192.168.2.1 255.255.255.0
!
router eigrp 1
 network 10.0.0.0 0.0.0.255
 network 192.168.1.0 0.0.0.255
 no auto-summary
!
access-list 101 permit gre 172.16.2.0 0.0.0.255 host
172.17.0.1
```

Le caratteristiche principali da notare per le configurazioni di spoke sono:

- L'indirizzo IP dell'interfaccia fisica esterna (ethernet0) è dinamico tramite DHCP.**indirizzo ip dhcp nomehost Spoke2**
- L'ACL crittografico (101) specifica una subnet come origine del proxy IPsec.**access-list 101 allow gre 172.16.2.0 0.0.0.255 host 172.17.0.1**
- Il comando seguente nella mappa crittografica di IPSec specifica che l'associazione di protezione verrà stabilita per host.**impostazione del livello di associazione di protezione per host**
- Tutti i tunnel fanno parte della stessa subnet, in quanto si connettono tutti tramite la stessa interfaccia GRE multipoint sul router hub.**indirizzo ip 10.0.0.2 255.255.255.0**

La combinazione di questi tre comandi rende superflua la configurazione dell'indirizzo IP dell'interfaccia fisica esterna dello spoke. Il proxy IPsec utilizzato sarà basato su host anziché su subnet.

La configurazione sui router spoke non dispone dell'indirizzo IP del router hub configurato, in quanto deve avviare il tunnel IPsec+GRE. Notate la somiglianza tra le configurazioni Spoke1 e Spoke2. Non solo queste due configurazioni sono simili, ma tutte le configurazioni del router spoke sono simili. Nella maggior parte dei casi, tutti gli spoke necessitano semplicemente di indirizzi IP univoci sulle loro interfacce, e il resto delle loro configurazioni sarà lo stesso. Questo consente di configurare e installare rapidamente molti router spoke.

I dati NHRP sono simili a quelli riportati di seguito nell'hub e nello spoke.

Router hub

```
Hub#show ip nhrp
   10.0.0.2/32 via 10.0.0.2, Tunnel0 created 01:25:18,
expire 00:03:51
     Type: dynamic, Flags: authoritative unique
registered
     NBMA address: 172.16.1.4
   10.0.0.3/32 via 10.0.0.3, Tunnel0 created 00:06:02,
expire 00:04:03
     Type: dynamic, Flags: authoritative unique
registered
     NBMA address: 172.16.2.10
   ...
   10.0.0.<n>/32 via 10.0.0.<n>, Tunnel0 created
00:06:00, expire 00:04:25
     Type: dynamic, Flags: authoritative unique
registered
     NBMA address: 172.16.<n>.41
```

Router Spoke1

```
Spoke1#sho ip nhrp
   10.0.0.1/32 via 10.0.0.1, Tunnel0 created 4d08h,
never expire
     Type: static, Flags: authoritative
     NBMA address: 172.17.0.1
```

Hub e spoke dinamici a più punti

La configurazione sui router spoke di cui sopra non si basa sulle funzionalità della soluzione DMVPN, quindi i router spoke possono eseguire versioni software Cisco IOS prima della 12.2(13)T. La configurazione sul router hub si basa sulle funzionalità DMVPN, quindi deve eseguire Cisco IOS versione 12.2(13)T o successive. Ciò consente una certa flessibilità nel decidere quando è necessario aggiornare i router spoke già distribuiti. Se i router spoke eseguono anche Cisco IOS versione 12.2(13)T o successive, è possibile semplificare la configurazione spoke come segue.

Router Spoke<n> (prima di Cisco IOS 12.2(13)T)

```
crypto map vpnmap1 10 IPsec-isakmp
  set peer 172.17.0.1
  set security-association level per-host
  set transform-set trans2
  match address 101
 !
 interface Tunnel0
  bandwidth 1000
  ip address 10.0.0.<n+1> 255.255.255.0
  ip mtu 1400
  ip nhrp authentication test
  ip nhrp map 10.0.0.1 172.17.0.1
  ip nhrp network-id 100000
  ip nhrp holdtime 300
  ip nhrp nhs 10.0.0.1
  delay 1000
  tunnel source Ethernet0
  tunnel destination 172.17.0.1
  tunnel key 100000
 !
 interface Ethernet0
  ip address dhcp hostname Spoke<n>
  crypto map vpnmap1
 !
 . . .
 !
 access-list 101 permit gre any host 172.17.0.1
```

Spoke<n> Router (dopo Cisco IOS 12.2(13)T)

```
crypto ipsec profile vpnprof
  set transform-set trans2
 !
 interface Tunnel0
  bandwidth 1000
  ip address 10.0.0.<n+1> 255.255.255.0
  ip mtu 1400
  ip nhrp authentication test
  ip nhrp map 10.0.0.1 172.17.0.1
  ip nhrp network-id 100000
  ip nhrp holdtime 300
  ip nhrp nhs 10.0.0.1
  delay 1000
  tunnel source Ethernet0
  tunnel destination 172.17.0.1
```

```
 tunnel key 100000
 tunnel protection ipsec profile vpnprof
!
interface Ethernet0
 ip address dhcp hostname Spoke<n>
!
```

Si noti che sono state effettuate le seguenti operazioni:

1. Il comando **crypto map vpnmap1 10 ipsec-isakmp** è stato rimosso e sostituito con il **profilo crypto ipsec vpnprof**.
2. Rimuovere il comando **crypto map vpnmap1** dalle interfacce Ethernet0 e inserire il comando **tunnel protection ipsec profile vpnprof** sull'interfaccia Tunnel0.
3. L'ACL crittografico è stato rimosso. **L'elenco degli accessi 101 consente al gre di qualsiasi host 172.17.0.1**.

In questo caso, gli indirizzi dei peer IPsec e i proxy vengono derivati automaticamente dall'**origine** del **tunnel** e dalla configurazione della **destinazione del tunnel**. I peer e i proxy sono i seguenti (come mostrato nell'output del comando **show crypto ipsec sa**):

```
...
local ident (addr/mask/prot/port):    (172.16.1.24/255.255.255.255/47/0)
remote ident (addr/mask/prot/port):   (172.17.0.1/255.255.255.255/47/0)
...
local crypto endpt.: 172.17.1.24, remote crypto endpt.:172.17.0.1
...
```

In breve, le seguenti configurazioni complete includono tutte le modifiche apportate fino a questo punto dalla Configurazione base (hub e spoke IPsec+GRE).

Router hub

```
version 12.3
 !
 hostname Hub
 !
 crypto isakmp policy 1
  authentication pre-share
 crypto isakmp key cisco47 address 0.0.0.0
 !
 crypto ipsec transform-set trans2 esp-des esp-md5-hmac
  mode transport
 !
 crypto ipsec profile vpnprof
  set transform-set trans2
 !
 interface Tunnel0
  bandwidth 1000
  ip address 10.0.0.1 255.255.255.0
  ip mtu 1400
  ip nhrp authentication test
  ip nhrp map multicast dynamic
  ip nhrp network-id 100000
  ip nhrp holdtime 600
  no ip split-horizon eigrp 1
  delay 1000
  tunnel source Ethernet0
```

```
 tunnel mode gre multipoint
 tunnel key 100000
 tunnel protection ipsec profile vpnprof
!
interface Ethernet0
 ip address 172.17.0.1 255.255.255.0
!
interface Ethernet1
 ip address 192.168.0.1 255.255.255.0
!
router eigrp 1
 network 10.0.0.0 0.0.0.255
 network 192.168.0.0 0.0.0.255
 no auto-summary
!
```

Nessuna modifica nella configurazione dell'hub.

```
version 12.3
!
hostname Spoke1
!
crypto isakmp policy 1
 authentication pre-share
crypto isakmp key cisco47 address 0.0.0.0 0.0.0.0
!
crypto ipsec transform-set trans2 esp-des esp-md5-hmac
 mode transport
!
crypto ipsec profile vpnprof
 set transform-set trans2
!
interface Tunnel0
 bandwidth 1000
 ip address 10.0.0.2 255.255.255.0
 ip mtu 1400
 ip nhrp authentication test
 ip nhrp map 10.0.0.1 172.17.0.1
 ip nhrp network-id 100000
 ip nhrp holdtime 300
 ip nhrp nhs 10.0.0.1
 delay 1000
 tunnel source Ethernet0
 tunnel destination 172.17.0.1
 tunnel key 100000
 tunnel protection ipsec profile vpnprof
!
interface Ethernet0
 ip address dhcp hostname Spoke2
!
interface Ethernet1
 ip address 192.168.1.1 255.255.255.0
!
router eigrp 1
 network 10.0.0.0 0.0.0.255
 network 192.168.1.0 0.0.0.255
 no auto-summary
!
```

Router Spoke2

```
version 12.3
 !
 hostname Spoke2
 !
 crypto isakmp policy 1
  authentication pre-share
 crypto isakmp key cisco47 address 0.0.0.0 0.0.0.0
 !
 crypto ipsec transform-set trans2 esp-des esp-md5-hmac
  mode transport
 !
 crypto ipsec profile vpnprof
  set transform-set trans2
 !
 interface Tunnel0
  bandwidth 1000
  ip address 10.0.0.3 255.255.255.0
  ip mtu 1400
  ip nhrp authentication test
  ip nhrp map 10.0.0.1 172.17.0.1
  ip nhrp network-id 100000
  ip nhrp holdtime 300
  ip nhrp nhs 10.0.0.1
  delay 1000
  tunnel source Ethernet0
  tunnel destination 172.17.0.1
  tunnel key 100000
  tunnel protection ipsec profile vpnprof
 !
 interface Ethernet0
  ip address dhcp hostname Spoke2
 !
 interface Ethernet1
  ip address 192.168.2.1 255.255.255.0
 !
 router eigrp 1
  network 10.0.0.0 0.0.0.255
  network 192.168.2.0 0.0.0.255
  no auto-summary
 !
```

VPN IPsec multipunto dinamica

I concetti e la configurazione illustrati in questa sezione mostrano le funzionalità complete di DMVPN. NHRP consente ai router spoke di imparare dinamicamente l'indirizzo di interfaccia fisica esterna degli altri router spoke nella rete VPN. Ciò significa che un router spoke disporrà di informazioni sufficienti per compilare dinamicamente un tunnel IPsec+mGRE direttamente ad altri router spoke. Questo è un vantaggio perché, se il traffico di dati spoke è stato inviato tramite il router hub, deve essere crittografato/decrittografato, aumentando due volte il ritardo e il carico sul router hub. Per utilizzare questa funzione, i router spoke devono essere commutati dalle interfacce tunnel GRE point-to-point (p-GRE) alle interfacce tunnel GRE multipoint (mGRE). Devono inoltre conoscere le (sotto)reti disponibili dietro gli altri spoke con un hop IP successivo dell'indirizzo IP del tunnel del router dell'altro spoke. I router spoke imparano queste (sotto)reti tramite il protocollo di routing IP dinamico in esecuzione sul tunnel IPsec+mGRE con l'hub.

Il protocollo di routing IP dinamico in esecuzione sul router hub può essere configurato in modo da riflettere i percorsi appresi da uno spoke verso l'esterno della stessa interfaccia a tutti gli altri spoke, ma l'hop successivo IP su questi percorsi sarà in genere il router hub, non il router spoke da cui l'hub ha appreso questo percorso.

Nota: il protocollo di routing dinamico viene eseguito solo sui collegamenti hub e spoke, non sui collegamenti spoke dinamici.

I protocolli di routing dinamico (RIP, OSPF ed EIGRP) devono essere configurati sul router dell'hub per annunciare i percorsi all'esterno dell'interfaccia del tunnel GRE e per impostare l'hop successivo IP sul router spoke di origine per i percorsi appresi da un spoke quando il percorso viene annunciato di nuovo agli altri spoke.

Di seguito sono riportati i requisiti per le configurazioni del protocollo di routing.

RIP

Ènecessario disattivare la divisione dell'orizzonte sull'interfaccia del tunnel GRE sull'hub. In caso contrario, RIP non annuncerà le route apprese tramite l'interfaccia mGRE e quindi la stessa interfaccia.

```
no ip split-horizon
```

Non sono necessarie altre modifiche. RIP utilizzerà automaticamente l'hop successivo IP originale sui percorsi che annuncia all'esterno della stessa interfaccia in cui ha appreso i percorsi.

EIGRP

Ènecessario disattivare la divisione dell'orizzonte sull'interfaccia del tunnel GRE sull'hub. In caso contrario, EIGRP non annuncerà le route apprese tramite l'interfaccia mGRE che si trovano nella stessa interfaccia.

```
no ip split-horizon eigrp
```

Per impostazione predefinita, l'EIGRP imposta l'hop successivo IP come router hub per i percorsi pubblicizzati, anche quando questi percorsi vengono annunciati all'esterno della stessa interfaccia in cui sono stati individuati. In questo caso, è necessario usare il seguente comando di configurazione per configurare EIGRP in modo che usi l'hop successivo IP originale quando si annunciano queste route.

```
no ip next-hop-self eigrp
```

Nota: il comando **no ip next-hop-self eigrp <as>** sarà disponibile a partire dalla versione 12.3(2) di Cisco IOS. Per le versioni Cisco IOS tra la 12.2(13)T e la 12.3(2), eseguire le operazioni seguenti:

- Se non si desidera utilizzare i tunnel dinamici spoke-to-spoke, il comando precedente non è necessario.
- Se si desiderano tunnel dinamici spoke, è necessario utilizzare la commutazione di contesto sull'interfaccia del tunnel sui router spoke.
- In caso contrario, sarà necessario utilizzare un protocollo di routing diverso sulla DMVPN.

OSPF

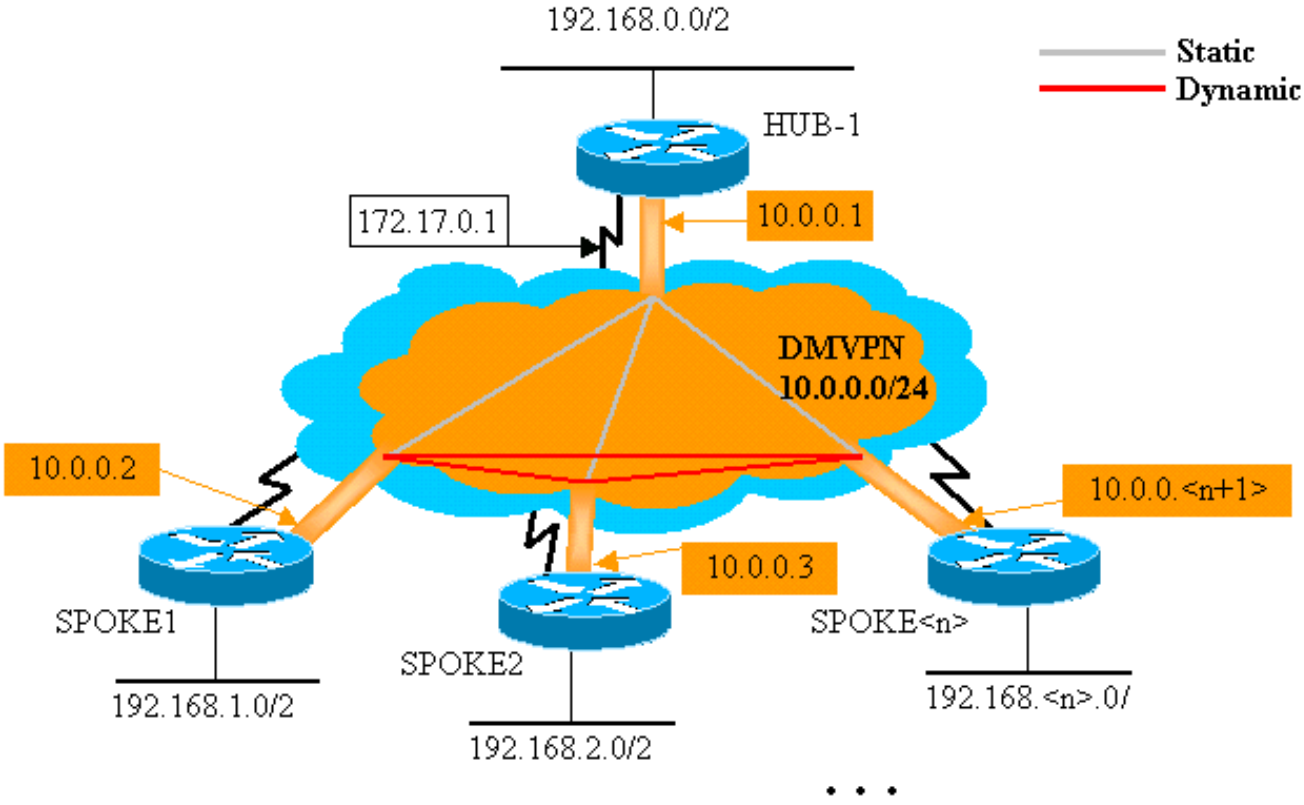
Poiché OSPF è un protocollo di routing allo stato di collegamento, non vi sono problemi di split-orizzonte. In genere, per le interfacce multipoint il tipo di rete OSPF viene configurato come point-to-multipoint, ma in questo caso OSPF aggiungerebbe route host alla tabella di routing sui router spoke. I percorsi di questi host causerebbero l'inoltro dei pacchetti destinati alle reti dietro i router di altri spoke tramite l'hub, piuttosto che l'inoltro diretto all'altro spoke. Per risolvere il problema, configurare il tipo di rete OSPF da trasmettere utilizzando il comando.

```
ip ospf network broadcast
```

Inoltre, è necessario verificare che il router hub sia il router designato (DR) per la rete IPsec+mGRE. A tale scopo, la priorità OSPF viene impostata su un valore maggiore di 1 sull'hub e di 0 sugli spoke.

- Hub: **priorità ip ospf 2**
- Raggi: **priorità ip ospf 0**

DMVPN Single Hub



Router hub

```
version 12.3
 !
 hostname Hub
 !
 crypto isakmp policy 1
  authentication pre-share
 crypto isakmp key cisco47 address 0.0.0.0
 !
 crypto ipsec transform-set trans2 esp-des esp-md5-hmac
  mode transport
 !
 crypto ipsec profile vpnprof
  set transform-set trans2
 !
 interface Tunnel0
  bandwidth 1000
  ip address 10.0.0.1 255.255.255.0
  ip mtu 1400
  ip nhrp authentication test
  ip nhrp map multicast dynamic
  ip nhrp network-id 100000
  ip nhrp holdtime 600
  ip ospf network broadcast
  ip ospf priority 2
  delay 1000
  tunnel source Ethernet0
  tunnel mode gre multipoint
  tunnel key 100000
  tunnel protection ipsec profile vpnprof
 !
 interface Ethernet0
```

```
 ip address 172.17.0.1 255.255.255.0
!
interface Ethernet1
 ip address 192.168.0.1 255.255.255.0
!
router ospf 1
 network 10.0.0.0 0.0.0.255 area 0
 network 192.168.0.0 0.0.0.255 area 0
!
```

L'unica modifica nella configurazione dell'hub è che OSPF è il protocollo di routing anziché EIGRP. Si noti che il tipo di rete OSPF è impostato su broadcast e la priorità è impostata su 2. Se si imposta il tipo di rete OSPF su broadcast, OSPF installerà le route per le reti dietro i router Spoke con un indirizzo IP dell'hop successivo come indirizzo del tunnel GRE per il router Spoke.

Router Spoke1

```
version 12.3
!
hostname Spoke1
!
crypto isakmp policy 1
 authentication pre-share
crypto isakmp key cisco47 address 0.0.0.0 0.0.0.0
!
crypto ipsec transform-set trans2 esp-des esp-md5-hmac
 mode transport
!
crypto ipsec profile vpnprof
 set transform-set trans2
!
interface Tunnel0
 bandwidth 1000
 ip address 10.0.0.2 255.255.255.0
 ip mtu 1400
 ip nhrp authentication test
 ip nhrp map multicast 172.17.0.1
 ip nhrp map 10.0.0.1 172.17.0.1
 ip nhrp network-id 100000
 ip nhrp holdtime 300
 ip nhrp nhs 10.0.0.1
 ip ospf network broadcast
 ip ospf priority 0
 delay 1000
 tunnel source Ethernet0
 tunnel mode gre multipoint
 tunnel key 100000
 tunnel protection ipsec profile vpnprof
!
interface Ethernet0
 ip address dhcp hostname Spoke1
!
interface Ethernet1
 ip address 192.168.1.1 255.255.255.0
!
router ospf 1
 network 10.0.0.0 0.0.0.255 area 0
 network 192.168.1.0 0.0.0.255 area 0
!
```

La configurazione sui router spoke è ora molto simile a quella sull'hub. Le differenze sono le seguenti:

- La priorità OSPF è impostata su 0. Impossibile consentire ai router spoke di diventare il DR per la rete NBMA (Non Broadcast Multiaccess) GRE. Solo il router hub dispone di connessioni statiche dirette a tutti i router spoke. Il DR deve avere accesso a tutti i membri della rete NBMA.
- Per il router hub sono configurati mapping unicast e multicast NHRP.

  ```
  ip nhrp map multicast 172.17.0.1
  ip nhrp map 10.0.0.1 172.17.0.1
  ```

  Nella configurazione precedente, il comando **ip nhrp map multicast ...** non era necessario perché il tunnel GRE era point-to-point. In questo caso, i pacchetti multicast verranno incapsulati automaticamente attraverso il tunnel fino alla singola destinazione possibile. Questo comando è necessario perché il tunnel GRE di spoke è stato modificato in multipunto e vi è più di una destinazione possibile.
- Quando il router spoke si accende, deve avviare la connessione del tunnel con l'hub, poiché il router hub non è configurato con alcuna informazione sui router spoke, ai quali potrebbero essere assegnati dinamicamente indirizzi IP. Anche i router spoke sono configurati con l'hub come NHRP NHS.
  `ip nhrp nhs 10.0.0.1`
  Con il comando precedente, il router spoke invia i pacchetti di registrazione NHRP tramite il tunnel GRE+IPsec al router hub a intervalli regolari. Questi pacchetti di registrazione forniscono le informazioni di mapping NHRP spoke necessarie al router hub per eseguire il tunnel dei pacchetti ai router spoke.

Router Spoke2

```
version 12.3
 !
 hostname Spoke2
 !
 crypto isakmp policy 1
  authentication pre-share
 crypto isakmp key cisco47 address 0.0.0.0 0.0.0.0
 !
 crypto ipsec transform-set trans2 esp-des esp-md5-hmac
  mode transport
 !
 crypto ipsec profile vpnprof
  set transform-set trans2
 !
 interface Tunnel0
  bandwidth 1000
  ip address 10.0.0.3 255.255.255.0
  ip mtu 1400
  ip nhrp authentication test
  ip nhrp map multicast 172.17.0.1
  ip nhrp map 10.0.0.1 172.17.0.1
  ip nhrp network-id 100000
  ip nhrp holdtime 300
  ip nhrp nhs 10.0.0.1
  ip ospf network broadcast
  ip ospf priority 0
  delay 1000
```

```
 tunnel source Ethernet0
 tunnel mode gre multipoint
 tunnel key 100000
 tunnel protection ipsec profile vpnprof
!
interface Ethernet0
 ip address dhcp hostname Spoke1
!
interface Ethernet1
 ip address 192.168.3.1 255.255.255.0
!
router ospf 1
 network 10.0.0.0 0.0.0.255 area 0
 network 192.168.2.0 0.0.0.255 area 0
!
```

Router Spoke<n>

```
version 12.3
!
hostname Spoke<n>
!
crypto isakmp policy 1
 authentication pre-share
crypto isakmp key cisco47 address 0.0.0.0 0.0.0.0
!
crypto ipsec transform-set trans2 esp-des esp-md5-hmac
 mode transport
!
crypto ipsec profile vpnprof
 set transform-set trans2
!
interface Tunnel0
 bandwidth 1000
 ip address 10.0.0.<n+1> 255.255.255.0
 ip mtu 1400
 ip nhrp authentication test
 ip nhrp map multicast 172.17.0.1
 ip nhrp map 10.0.0.1 172.17.0.1
 ip nhrp network-id 100000
 ip nhrp holdtime 300
 ip nhrp nhs 10.0.0.1
 ip ospf network broadcast
 ip ospf priority 0
 delay 1000
 tunnel source Ethernet0
 tunnel mode gre multipoint
 tunnel key 100000
 tunnel protection ipsec profile vpnprof
!
interface Ethernet0
 ip address dhcp hostname Spoke<n>
!
interface Ethernet1
 ip address 192.168.<n>.1 255.255.255.0
!
router ospf 1
 network 10.0.0.0 0.0.0.255 area 0
 network 192.168.
```

```
!
```

Si noti che le configurazioni di tutti i router spoke sono molto simili. Le uniche differenze sono gli indirizzi IP sulle interfacce locali. Ciò è utile quando si distribuisce un numero elevato di router spoke. Tutti i router spoke possono essere configurati in modo identico e solo gli indirizzi dell'interfaccia IP locale devono essere aggiunti.

A questo punto, osservare le tabelle di routing e le tabelle di mapping NHRP sui router Hub, Spoke1 e Spoke2 per verificare le condizioni iniziali (subito dopo l'accensione dei router Spoke1 e Spoke2) e le condizioni successive alla creazione di un collegamento dinamico tra Spoke1 e Spoke2.

Condizioni iniziali

Informazioni sul router hub

```
Hub#show ip route
      172.17.0.0/24 is subnetted, 1 subnets
 C       172.17.0.0 is directly connected, Ethernet0
      10.0.0.0/24 is subnetted, 1 subnets
 C       10.0.0.0 is directly connected, Tunnel0
 C    192.168.0.0/24 is directly connected, Ethernet1
 O    192.168.1.0/24 [110/2] via 10.0.0.2, 00:19:53,
Tunnel0
 O    192.168.2.0/24 [110/2] via 10.0.0.3, 00:19:53,
Tunnel0
 Hub#show ip nhrp
 10.0.0.2/32 via 10.0.0.2, Tunnel0 created 00:57:27,
expire    00:04:13
   Type: dynamic, Flags: authoritative unique registered
   NBMA address: 172.16.1.24
 10.0.0.3/32 via 10.0.0.3, Tunnel0 created 07:11:25,
expire    00:04:33
   Type: dynamic, Flags: authoritative unique registered
   NBMA address: 172.16.2.75
 Hub#show crypto engine connection active
  ID  Interface   IP-Address    State Algorithm
Encrypt Decrypt
  204 Ethernet0   172.17.0.1     set  HMAC_SHA+DES_56_CB
0       0
  205 Ethernet0   172.17.0.1     set  HMAC_SHA+DES_56_CB
0       0
 2628 Tunnel0     10.0.0.1       set  HMAC_MD5
0     402
 2629 Tunnel0     10.0.0.1       set  HMAC_MD5
357       0
 2630 Tunnel0     10.0.0.1       set  HMAC_MD5
0     427
 2631 Tunnel0     10.0.0.1       set  HMAC_MD5
308       0
```

Informazioni su Spoke1 Router

```
Spoke1#show ip route
      172.16.0.0/24 is subnetted, 1 subnets
 C       172.16.1.24 is directly connected, Ethernet0
```

```
      10.0.0.0/24 is subnetted, 1 subnets
 C       10.0.0.0 is directly connected, Tunnel0
 O    192.168.0.0/24 [110/2] via 10.0.0.1, 00:31:46,
Tunnel0
 C    192.168.1.0/24 is directly connected, Ethernet1
 O    192.168.2.0/24 [110/2] via 10.0.0.3, 00:31:46,
Tunnel0
 Spoke1#show ip nhrp
 10.0.0.1/32 via 10.0.0.1, Tunnel0 created 01:42:00,
never expire
   Type: static, Flags: authoritative used
   NBMA address: 172.17.0.1
 Spoke1#show crypto engine connection active
  ID  Interface   IP-Address    State Algorithm
Encrypt Decrypt
    2 Ethernet0   172.16.1.24    set  HMAC_SHA+DES_56_CB
0       0
 2064 Tunnel0     10.0.0.2       set  HMAC_MD5
0     244
 2065 Tunnel0     10.0.0.2       set  HMAC_MD5
276       0
```

Informazioni router Spoke2

```
Spoke2#show ip route
      172.16.0.0/24 is subnetted, 1 subnets
 C       172.16.2.0 is directly connected, Ethernet0
      10.0.0.0/24 is subnetted, 1 subnets
 C       10.0.0.0 is directly connected, Tunnel0
 O    192.168.0.0/24 [110/2] via 10.0.0.1, 00:38:52,
Tunnel0
 O    192.168.1.0/24 [110/2] via 10.0.0.2, 00:38:52,
Tunnel0
 C    192.168.2.0/24 is directly connected, Ethernet1
 Spoke2#show ip nhrp
 10.0.0.1/32 via 10.0.0.1, Tunnel0 created 01:32:10,
never expire
   Type: static, Flags: authoritative used
   NBMA address: 172.17.0.1
 Spoke2#show crypto engine connection active
  ID  Interface   IP-Address    State Algorithm
Encrypt Decrypt
   17 Ethernet0   172.16.2.75    set  HMAC_SHA+DES_56_CB
0       0
 2070 Tunnel0     10.0.0.3       set  HMAC_MD5
0     279
 2071 Tunnel0     10.0.0.3       set  HMAC_MD5
316       0
```

A questo punto, è possibile eseguire il ping tra le versioni 192.168.1.2 e 192.168.2.3. Questi indirizzi sono destinati agli host dietro i router Spoke1 e Spoke2, rispettivamente. La seguente sequenza di eventi ha luogo per compilare il tunnel GRE+IPsec spoke diretto.

1. Il router Spoke1 riceve il pacchetto ping con destinazione 192.168.2.3. Cerca questa destinazione nella tabella di routing e trova la necessità di inoltrare il pacchetto dall'interfaccia Tunnel0 all'IP nexthop, 10.0.0.3.
2. Il router Spoke1 controlla la tabella di mapping NHRP per la destinazione 10.0.0.3 e rileva che non esiste una voce. Il router Spoke1 crea un pacchetto di richiesta di risoluzione NHRP e lo invia al proprio NHS (il router hub).

3. Il router hub controlla la tabella di mapping NHRP per la destinazione 10.0.0.3 e rileva che è mappata all'indirizzo 172.16.2.75. Il router hub crea un pacchetto di risposta alla risoluzione NHRP e lo invia al router Spoke1.
4. Il router Spoke1 riceve la risposta alla risoluzione NHRP e immette il mapping 10.0.0.3 —>172.16.2.75 nella relativa tabella di mapping NHRP. L'aggiunta del mapping NHRP attiva IPsec per avviare un tunnel IPsec con il peer 172.16.2.75.
5. Il router Spoke1 avvia ISAKMP con 172.16.2.75 e negozia le associazioni di protezione ISAKMP e IPsec. Il proxy IPsec viene derivato dal comando Tunnel0 **tunnel source <address>** e dal mapping NHRP.

```
local ident (addr/mask/prot/port): (172.16.1.24/255.255.255.255/47/0)
remote ident (addr/mask/prot/port): (172.16.2.75/255.255.255.255/47/0)
```

6. Al termine della creazione del tunnel IPsec, tutti gli altri pacchetti di dati nella subnet 192.168.2.0/24 vengono inviati direttamente a Spoke2.
7. Dopo aver inoltrato all'host un pacchetto destinato alla versione 192.168.2.3, l'host invierà un pacchetto di ritorno alla versione 192.168.1.2. Quando il router Spoke2 riceve il pacchetto destinato alla versione 192.168.1.2, cerca la destinazione nella tabella di routing e scopre che deve inoltrare il pacchetto dall'interfaccia Tunnel0 all'hop successivo IP, la versione 10.0.0.2.
8. Il router Spoke2 controlla la tabella di mapping NHRP per la destinazione 10.0.0.2 e rileva che non esiste una voce. Il router Spoke2 crea un pacchetto di richiesta di risoluzione NHRP e lo invia al proprio NHS (il router hub).
9. Il router hub controlla la tabella di mapping NHRP per la destinazione 10.0.0.2 e rileva che è mappata all'indirizzo 172.16.1.24. Il router hub crea un pacchetto di risposta alla risoluzione NHRP e lo invia al router Spoke2.
10. Il router Spoke2 riceve la risposta alla risoluzione NHRP e immette il mapping 10.0.0.2 —> 172.16.1.24 nella relativa tabella di mapping NHRP. L'aggiunta del mapping NHRP attiva IPsec per avviare un tunnel IPsec con peer 172.16.1.24, ma esiste già un tunnel IPsec con peer 172.16.1.24, quindi non è necessario eseguire altre operazioni.
11. Spoke1 e Spoke2 sono ora in grado di inoltrare i pacchetti direttamente tra loro. Se il mapping NHRP non è stato utilizzato per l'inoltro dei pacchetti per il periodo di sospensione, verrà eliminato. L'eliminazione della voce di mapping NHRP attiverà IPSec per eliminare le associazioni di protezione IPsec per questo collegamento diretto.

Condizioni dopo la creazione di un collegamento dinamico tra Spoke1 e Spoke2

Informazioni su Spoke1 Router

```
Spoke1#show ip nhrp
 10.0.0.1/32 via 10.0.0.1, Tunnel0 created 02:34:16,
never expire
   Type: static, Flags: authoritative used
   NBMA address: 172.17.0.1
 10.0.0.3/32 via 10.0.0.3, Tunnel0 created 00:00:05,
expire 00:03:35
   Type: dynamic, Flags: router unique used
   NBMA address: 172.16.2.75
 Spoke1#show crypto engine connection active
  ID  Interface   IP-Address    State Algorithm
```

```
Encrypt Decrypt
    2 Ethernet0   172.16.1.24    set  HMAC_SHA+DES_56_CB
0       0
    3 Ethernet0   172.16.1.24    set  HMAC_SHA+DES_56_CB
0       0
 2064 Tunnel0     10.0.0.2       set  HMAC_MD5
0     375
 2065 Tunnel0     10.0.0.2       set  HMAC_MD5
426       0
 2066 Tunnel0     10.0.0.2       set  HMAC_MD5
0      20
 2067 Tunnel0     10.0.0.2       set  HMAC_MD5
19        0
```

Informazioni router Spoke2

```
Spoke2#show ip nhrp
 10.0.0.1/32 via 10.0.0.1, Tunnel0 created 02:18:25,
never expire
   Type: static, Flags: authoritative used
   NBMA address: 172.17.0.1
 10.0.0.2/32 via 10.0.0.2, Tunnel0 created 00:00:24,
expire 00:04:35
   Type: dynamic, Flags: router unique used
   NBMA address: 172.16.1.24
 Spoke2#show crypto engine connection active
  ID  Interface   IP-Address    State Algorithm
Encrypt Decrypt
   17 Ethernet0   172.16.2.75    set  HMAC_SHA+DES_56_CB
0       0
   18 Ethernet0   172.16.2.75    set  HMAC_SHA+DES_56_CB
0       0
 2070 Tunnel0     10.0.0.3       set  HMAC_MD5
0     407
 2071 Tunnel0     10.0.0.3       set  HMAC_MD5
460       0
 2072 Tunnel0     10.0.0.3       set  HMAC_MD5
0      19
 2073 Tunnel0     10.0.0.3       set  HMAC_MD5
20        0
```

Dall'output sopra riportato, si può vedere che Spoke1 e Spoke2 hanno ricevuto reciprocamente i mapping NHRP dal router Hub e hanno creato e utilizzato un tunnel GRE+IPsec. I mapping NHRP scadranno dopo cinque minuti (il valore corrente di tempo di attesa NHRP = 300 secondi). Se i mapping NHRP vengono utilizzati nell'ultimo minuto prima della scadenza, verranno inviate una richiesta e una risposta di risoluzione NHRP per aggiornare la voce prima che venga eliminata. In caso contrario, il mapping NHRP verrà eliminato e IPsec verrà attivato per cancellare le associazioni di protezione IPsec.

VPN IPsec multipoint dinamica con hub doppi

Con alcune linee di configurazione aggiuntive ai router spoke è possibile configurare router hub doppi (o multipli) per la ridondanza. Esistono due modi per configurare le DMVPN con hub doppio.

- Una singola rete DMVPN con ciascun spoke che utilizza un'unica interfaccia del tunnel GRE multipoint e punta a due hub diversi come server NHS (Next-Hop-Server).I router hub avranno solo un'unica interfaccia del tunnel GRE multipoint.

- Reti DMVPN doppie con ciascun spoke avente due interfacce del tunnel GRE (point-to-point o multipoint) e ciascun tunnel GRE connesso a un router hub diverso.Anche in questo caso, i router hub avranno solo un'unica interfaccia del tunnel GRE multipoint.

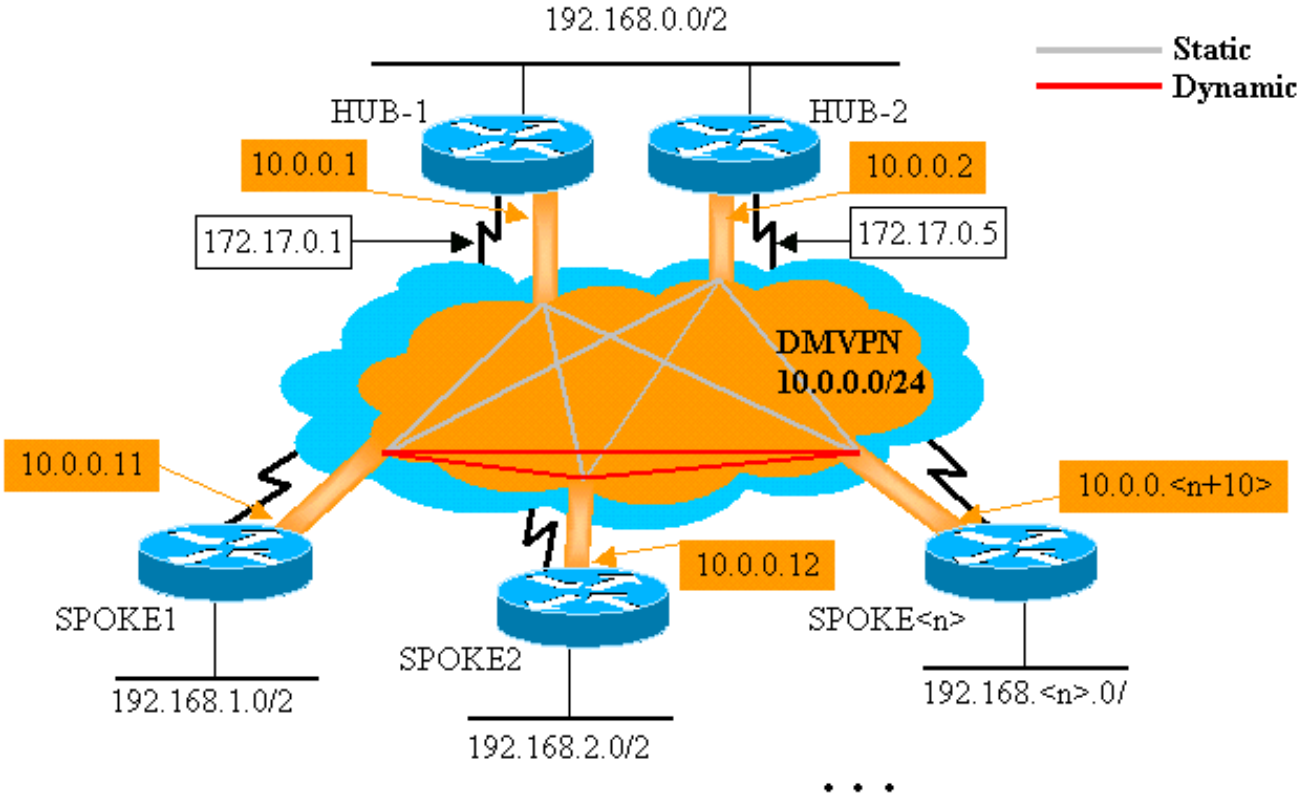
Negli esempi seguenti verrà esaminata la configurazione di questi due diversi scenari per le DMVPN con hub doppio. In entrambi i casi, le differenze evidenziate sono relative alla configurazione dell'hub singolo DMVPN.

Doppio hub - layout DMVPN singolo

L'hub doppio con un unico layout DMVPN è abbastanza facile da configurare, ma non offre lo stesso controllo sul routing tramite DMVPN di cui dispone l'hub doppio con layout DMVPN doppio. In questo caso, l'idea è di avere un'unica "cloud" DMVPN con tutti gli hub (due in questo caso) e tutti i spoke connessi a questa singola subnet ("cloud"). I mapping NHRP statici dagli spoke agli hub definiscono i collegamenti IPsec+mGRE statici su cui verrà eseguito il protocollo di routing dinamico. Il protocollo di routing dinamico non verrà eseguito sui collegamenti IPsec+mGRE dinamici tra gli spoke. Poiché i router spoke eseguono il routing dei router adiacenti con i router hub sulla stessa interfaccia del tunnel GRE, non è possibile utilizzare differenze di collegamento o interfacce (come metriche, costi, ritardo o larghezza di banda) per modificare le metriche del protocollo di routing dinamico in modo da preferire un hub all'altro hub quando sono entrambi attivi. Se questa preferenza è necessaria, è necessario usare tecniche interne alla configurazione del protocollo di routing. Per questo motivo, potrebbe essere preferibile utilizzare EIGRP o RIP anziché OSPF per il protocollo di routing dinamico.

Nota: questo problema si verifica in genere solo se i router dell'hub sono ubicati nello stesso percorso. Quando non si trovano nello stesso luogo, il normale routing dinamico finirà probabilmente per preferire il router hub corretto, anche se la rete di destinazione può essere raggiunta tramite uno dei router hub.

Doppio hub - layout DMVPN singolo



Router hub

```
version 12.3
 !
 hostname Hub1
 !
 crypto isakmp policy 1
  authentication pre-share
 crypto isakmp key cisco47 address 0.0.0.0
 !
 crypto ipsec transform-set trans2 esp-des esp-md5-hmac
  mode transport
 !
 crypto ipsec profile vpnprof
  set transform-set trans2
 !
 interface Tunnel0
  bandwidth 1000
  ip address 10.0.0.1 255.255.255.0
  ip mtu 1400
  ip nhrp authentication test
  ip nhrp map multicast dynamic
  ip nhrp network-id 100000
  ip nhrp holdtime 600
  ip ospf network broadcast
  ip ospf priority 2
  delay 1000
  tunnel source Ethernet0
  tunnel mode gre multipoint
  tunnel key 100000
  tunnel protection ipsec profile vpnprof
```

```
 !
 interface Ethernet0
  ip address 172.17.0.1 255.255.255.0
 !
 interface Ethernet1
  ip address 192.168.0.1 255.255.255.0
 !
 router ospf 1
  network 10.0.0.0 0.0.0.255 area 1
  network 192.168.0.0 0.0.0.255 area 0
 !
```

Router Hub2

```
version 12.3
 !
 hostname Hub2
 !
 crypto isakmp policy 1
  authentication pre-share
 crypto isakmp key cisco47 address 0.0.0.0
 !
 crypto ipsec transform-set trans2 esp-des esp-md5-hmac
  mode transport
 !
 crypto ipsec profile vpnprof
  set transform-set trans2
 !
 interface Tunnel0
  bandwidth 900
  ip address 10.0.0.2 255.255.255.0
  ip mtu 1400
  ip nhrp authentication test
  ip nhrp map 10.0.0.1 172.17.0.1
  ip nhrp map  multicast 172.17.0.1
  ip nhrp map multicast dynamic
  ip nhrp network-id 100000
  ip nhrp holdtime 600
  ip nhrp nhs 10.0.0.1
  ip ospf network broadcast
  ip ospf priority 1
  delay 1000
  tunnel source Ethernet0
  tunnel mode gre multipoint
  tunnel key 100000
  tunnel protection ipsec profile vpnprof
 !
 interface Ethernet0
  ip address 172.17.0.5 255.255.255.0
 !
 interface Ethernet1
  ip address 192.168.0.2 255.255.255.0
 !
 router ospf 1
  network 10.0.0.0 0.0.0.255 area 1
  network 192.168.0.0 0.0.0.255 area 0
 !
```

L'unica modifica nella configurazione dell'hub 1 consiste nel modificare l'OSPF in modo da utilizzare due aree. L'area 0 viene utilizzata per la rete dietro i due hub, mentre l'area 1 viene utilizzata per la rete DMVPN e le reti dietro i router spoke. L'OSPF può utilizzare una singola area,

ma sono state utilizzate due aree per dimostrare la configurazione di più aree OSPF.

La configurazione di Hub2 è fondamentalmente la stessa della configurazione di Hub1 con le modifiche all'indirizzo IP appropriate. L'unica differenza principale è che Hub2 è anche un spoke (o client) di Hub1, rendendo Hub1 l'hub primario e Hub2 l'hub secondario. In questo modo, Hub2 è un router adiacente OSPF con Hub1 sul tunnel GRE. Poiché l'hub 1 è il DR OSPF, deve disporre di una connessione diretta con tutti gli altri router OSPF tramite l'interfaccia GRE (rete NBMA). Senza il collegamento diretto tra Hub1 e Hub2, l'Hub2 non parteciperebbe al routing OSPF quando anche l'Hub1 è attivo. Quando Hub1 non è attivo, Hub2 sarà il DR OSPF per la rete DMVPN (NBMA). Quando l'hub 1 ritorna disponibile, diventerà il DR OSPF per la VPN DMVPN.

I router dietro l'hub1 e l'hub2 utilizzeranno l'hub1 per inviare i pacchetti alle reti spoke perché la larghezza di banda per l'interfaccia del tunnel GRE è impostata su 1000 Kb/sec rispetto a 900 Kb/sec sull'hub2. Al contrario, i router spoke invieranno i pacchetti per le reti dietro i router hub a entrambi gli hub1 e all'hub2, poiché su ogni router spoke è presente solo un'unica interfaccia del tunnel GRE e saranno presenti due route a costo uguale. Se si utilizza il bilanciamento del carico per pacchetto, i pacchetti potrebbero non essere ordinati.

Router Spoke1

```
version 12.3
 !
 hostname Spoke1
 !
 crypto isakmp policy 1
  authentication pre-share
 crypto isakmp key cisco47 address 0.0.0.0 0.0.0.0
 !
 crypto ipsec transform-set trans2 esp-des esp-md5-hmac
  mode transport
 !
 crypto ipsec profile vpnprof
  set transform-set trans2
 !
 interface Tunnel0
  bandwidth 1000
  ip address 10.0.0.11 255.255.255.0
  ip mtu 1400
  ip nhrp authentication test
  ip nhrp map multicast 172.17.0.1
  ip nhrp map 10.0.0.1 172.17.0.1
  ip nhrp map multicast 172.17.0.5
  ip nhrp map 10.0.0.2 172.17.0.5
  ip nhrp network-id 100000
  ip nhrp holdtime 300
  ip nhrp nhs 10.0.0.1
  ip nhrp nhs 10.0.0.2
  ip ospf network broadcast
  ip ospf priority 0
  delay 1000
  tunnel source Ethernet0
  tunnel mode gre multipoint
  tunnel key 100000
  tunnel protection ipsec profile vpnprof
 !
 interface Ethernet0
  ip address dhcp hostname Spoke1
 !
```

```
 interface Ethernet1
  ip address 192.168.1.1 255.255.255.0
 !
 router ospf 1
  network 10.0.0.0 0.0.0.255 area 1
  network 192.168.1.0 0.0.0.255 area 1
 !
```

Le differenze nella configurazione sui router spoke sono le seguenti:

- Nella nuova configurazione, lo spoke è configurato con mapping NHRP statici per Hub2 e Hub2 viene aggiunto come server dell'hop successivo.Originale:

  ```
  ip nhrp map multicast 172.17.0.1
  ip nhrp map 10.0.0.1 172.17.0.1
  ip nhrp nhs 10.0.0.1
  ```

  Nuovo:

  ```
  ip nhrp map multicast 172.17.0.1
  ip nhrp map 10.0.0.1 172.17.0.1
  ip nhrp map multicast 172.17.0.5
  ip nhrp map 10.0.0.2 172.17.0.5
  ip nhrp nhs 10.0.0.1
  ip nhrp nhs 10.0.0.2
  ```

- Le aree OSPF sui router spoke sono state modificate in area 1.

Tenere presente che definendo la mappatura NHRP statica e NHS su un router spoke per un hub, si eseguirà il protocollo di routing dinamico su questo tunnel. Definisce il routing hub e spoke per la rete adiacente. L'hub 2 è un hub per tutti gli spoke ed è anche uno spoke per Hub1. Ciò semplifica la progettazione, la configurazione e la modifica di reti hub e spoke multilivello quando si utilizza la soluzione DMVPN.

Router Spoke2

```
version 12.3
 !
 hostname Spoke2
 !
 crypto isakmp policy 1
  authentication pre-share
 crypto isakmp key cisco47 address 0.0.0.0 0.0.0.0
 !
 crypto ipsec transform-set trans2 esp-des esp-md5-hmac
  mode transport
 !
 crypto ipsec profile vpnprof
  set transform-set trans2
 !
 interface Tunnel0
  bandwidth 1000
  ip address 10.0.0.12 255.255.255.0
  ip mtu 1400
  ip nhrp authentication test
  ip nhrp map multicast 172.17.0.1
  ip nhrp map 10.0.0.1 172.17.0.1
  ip nhrp map multicast 172.17.0.5
  ip nhrp map 10.0.0.2 172.17.0.5
```

```
 ip nhrp network-id 100000
 ip nhrp holdtime 300
 ip nhrp nhs 10.0.0.1
 ip nhrp nhs 10.0.0.2
 ip ospf network broadcast
 ip ospf priority 0
 delay 1000
 tunnel source Ethernet0
 tunnel mode gre multipoint
 tunnel key 100000
 tunnel protection ipsec profile vpnprof
!
interface Ethernet0
 ip address dhcp hostname Spoke1
!
interface Ethernet1
 ip address 192.168.2.1 255.255.255.0
!
router ospf 1
 network 10.0.0.0 0.0.0.255 area 0
 network 192.168.2.0 0.0.0.255 area 0
!
```

Router Spoke<n>

```
version 12.3
!
hostname Spoke<n>
!
crypto isakmp policy 1
 authentication pre-share
crypto isakmp key cisco47 address 0.0.0.0 0.0.0.0
!
crypto ipsec transform-set trans2 esp-des esp-md5-hmac
 mode transport
!
crypto ipsec profile vpnprof
 set transform-set trans2
!
interface Tunnel0
 bandwidth 1000
 ip address 10.0.0.<n+10> 255.255.255.0
 ip mtu 1400
 ip nhrp authentication test
 ip nhrp map multicast 172.17.0.1
 ip nhrp map 10.0.0.1 172.17.0.1
 ip nhrp map multicast 172.17.0.5
 ip nhrp map 10.0.0.2 172.17.0.5
 ip nhrp network-id 100000
 ip nhrp holdtime 300
 ip nhrp nhs 10.0.0.1
 ip nhrp nhs 10.0.0.2
 ip ospf network broadcast
 ip ospf priority 0
 delay 1000
 tunnel source Ethernet0
 tunnel mode gre multipoint
 tunnel key 100000
 tunnel protection ipsec profile vpnprof
!
interface Ethernet0
 ip address dhcp hostname Spoke<x>
```

```
!
interface Ethernet1
 ip address 192.168.<n>.1 255.255.255.0
!
router ospf 1
 network 10.0.0.0 0.0.0.255 area 0
 network 192.168.




!
```

A questo punto, è possibile esaminare le tabelle di routing, le tabelle di mapping NHRP e le connessioni IPsec sui router Hub1, Hub2, Spoke1 e Spoke2 per verificare le condizioni iniziali (subito dopo l'accensione dei router Spoke1 e Spoke2).

Condizioni iniziali e modifiche

Informazioni sul router Hub1

```
Hub1#show ip route
      172.17.0.0/24 is subnetted, 1 subnets
 C       172.17.0.0 is directly connected, Ethernet0
      10.0.0.0/24 is subnetted, 1 subnets
 C       10.0.0.0 is directly connected, Tunnel0
 C    192.168.0.0/24 is directly connected, Ethernet1
 O    192.168.1.0/24 [110/2] via 10.0.0.11, 00:02:17,
Tunnel0
 O    192.168.2.0/24 [110/2] via 10.0.0.12, 00:02:17,
Tunnel0
 Hub1#show ip nhrp
 10.0.0.2/32 via 10.0.0.2, Tunnel0 created 1w3d, expire
00:03:15
   Type: dynamic, Flags: authoritative unique registered
   NBMA address: 172.17.0.5
 10.0.0.11/32 via 10.0.0.11, Tunnel0 created 1w3d,
expire 00:03:49
   Type: dynamic, Flags: authoritative unique registered
   NBMA address: 172.16.1.24
 10.0.0.12/32 via 10.0.0.12, Tunnel0 created 1w3d,
expire 00:04:06
   Type: dynamic, Flags: authoritative unique registered
   NBMA address: 172.16.2.75
 Hub1#show crypto engine connection active
  ID  Interface   IP-Address    State Algorithm
Encrypt Decrypt
    4 Ethernet0   172.17.0.1     set  HMAC_SHA+DES_56_CB
0       0
    5 Ethernet0   172.17.0.1     set  HMAC_SHA+DES_56_CB
0       0
    6 Ethernet0   172.17.0.1     set  HMAC_SHA+DES_56_CB
0       0
 3532 Tunnel0     10.0.0.1       set  HMAC_MD5+DES_56_CB
0     232
 3533 Tunnel0     10.0.0.1       set  HMAC_MD5+DES_56_CB
212       0
 3534 Tunnel0     10.0.0.1       set  HMAC_MD5+DES_56_CB
0      18
```

```
 3535 Tunnel0     10.0.0.1        set  HMAC_MD5+DES_56_CB
17       0
 3536 Tunnel0     10.0.0.1        set  HMAC_MD5+DES_56_CB
0        7
 3537 Tunnel0     10.0.0.1        set  HMAC_MD5+DES_56_CB
7        0
```

Informazioni router Hub2

```
Hub2#show ip route
      172.17.0.0/24 is subnetted, 1 subnets
 C       172.17.0.0 is directly connected, Ethernet0
      10.0.0.0/24 is subnetted, 1 subnets
 C       10.0.0.0 is directly connected, Tunnel0
 C    192.168.0.0/24 is directly connected, Ethernet1
 O    192.168.1.0/24 [110/2] via 10.0.0.11, 00:29:15,
Tunnel0
 O    192.168.2.0/24 [110/2] via 10.0.0.12, 00:29:15,
Tunnel0
 Hub2#show ip nhrp
 10.0.0.1/32 via 10.0.0.1, Tunnel0 created 1w3d, never
expire
   Type: static, Flags: authoritative used
   NBMA address: 172.17.0.1
 10.0.0.11/32 via 10.0.0.11, Tunnel0 created 1w3d,
expire 00:03:15
   Type: dynamic, Flags: authoritative unique registered
   NBMA address: 172.16.1.24
 10.0.0.12/32 via 10.0.0.12, Tunnel0 created 00:46:17,
expire 00:03:51
   Type: dynamic, Flags: authoritative unique registered
   NBMA address: 172.16.2.75
 Hub2#show crypto engine connection active
   ID Interface   IP-Address    State Algorithm
Encrypt Decrypt
    4 Ethernet0   171.17.0.5     set  HMAC_SHA+DES_56_CB
0        0
    5 Ethernet0   171.17.0.5     set  HMAC_SHA+DES_56_CB
0        0
    6 Ethernet0   171.17.0.5     set  HMAC_SHA+DES_56_CB
0        0
 3520 Tunnel0     10.0.0.2        set  HMAC_MD5+DES_56_CB
0      351
 3521 Tunnel0     10.0.0.2        set  HMAC_MD5+DES_56_CB
326        0
 3522 Tunnel0     10.0.0.2        set  HMAC_MD5+DES_56_CB
0      311
 3523 Tunnel0     10.0.0.2        set  HMAC_MD5+DES_56_CB
339        0
 3524 Tunnel0     10.0.0.2        set  HMAC_MD5+DES_56_CB
0       25
 3525 Tunnel0     10.0.0.2        set  HMAC_MD5+DES_56_CB
22         0
```

Informazioni su Spoke1 Router

```
Spoke1#show ip route
      172.16.0.0/24 is subnetted, 1 subnets
 C       172.16.1.0 is directly connected, Ethernet0
      10.0.0.0/24 is subnetted, 1 subnets
 C       10.0.0.0 is directly connected, Tunnel0
```

```
 O IA 192.168.0.0/24 [110/11] via 10.0.0.1, 00:39:31,
Tunnel0
                     [110/11] via 10.0.0.2, 00:39:31,
Tunnel0
 C    192.168.1.0/24 is directly connected, Ethernet1
 O    192.168.2.0/24 [110/2] via 10.0.0.12, 00:37:58,
Tunnel0
 Spoke1#show ip nhrp
 10.0.0.1/32 via 10.0.0.1, Tunnel0 created 00:56:40,
never expire
   Type: static, Flags: authoritative used
   NBMA address: 172.17.0.1
 10.0.0.2/32 via 10.0.0.2, Tunnel0 created 00:56:40,
never expire
   Type: static, Flags: authoritative used
   NBMA address: 172.17.0.5
 Spoke1#show crypto engine connection active
  ID  Interface   IP-Address    State Algorithm
Encrypt Decrypt
    1 Ethernet0   172.16.1.24    set  HMAC_SHA+DES_56_CB
0       0
    2 Ethernet0   172.16.1.24    set  HMAC_SHA+DES_56_CB
0       0
 2010 Tunnel0     10.0.0.11      set  HMAC_MD5+DES_56_CB
0     171
 2011 Tunnel0     10.0.0.11      set  HMAC_MD5+DES_56_CB
185       0
 2012 Tunnel0     10.0.0.11      set  HMAC_MD5+DES_56_CB
0      12
 2013 Tunnel0     10.0.0.11      set  HMAC_MD5+DES_56_CB
13       0
```

Informazioni router Spoke2

```
Spoke2#show ip route
      172.16.0.0/24 is subnetted, 1 subnets
 C       172.16.2.0 is directly connected, Ethernet0
      10.0.0.0/24 is subnetted, 1 subnets
 C       10.0.0.0 is directly connected, Tunnel0
 O IA 192.168.0.0/24 [110/11] via 10.0.0.1, 00:57:56,
Tunnel0
                     [110/11] via 10.0.0.2, 00:57:56,
Tunnel0
 O    192.168.1.0/24 [110/2] via 10.0.0.11, 00:56:14,
Tunnel0
 C    192.168.2.0/24 is directly connected, Ethernet1
 Spoke2#show ip nhrp
 10.0.0.1/32 via 10.0.0.1, Tunnel0 created 5w6d, never
expire
   Type: static, Flags: authoritative used
   NBMA address: 172.17.0.1
 10.0.0.2/32 via 10.0.0.2, Tunnel0 created 6w6d, never
expire
   Type: static, Flags: authoritative used
   NBMA address: 172.17.0.5
 Spoke2#show crypto engine connection active
  ID  Interface   IP-Address    State Algorithm
Encrypt Decrypt
    2 Ethernet0   172.16.2.75    set  HMAC_SHA+DES_56_CB
0       0
    3 Ethernet0   172.16.2.75    set  HMAC_SHA+DES_56_CB
0       0
```

```
 3712 Tunnel0     10.0.0.12      set  HMAC_MD5+DES_56_CB
0     302
 3713 Tunnel0     10.0.0.12      set  HMAC_MD5+DES_56_CB
331       0
 3716 Tunnel0     10.0.0.12      set  HMAC_MD5+DES_56_CB
0     216
 3717 Tunnel0     10.0.0.12      set  HMAC_MD5+DES_56_CB
236       0
```

Èpossibile notare un paio di problemi interessanti relativi alle tabelle di routing su Hub1, Hub2, Spoke1 e Spoke2:

- Entrambi i router hub hanno route a parità di costo verso le reti dietro i router spoke.Hub1:
  ```
  O     192.168.1.0/24 [110/2] via 10.0.0.11, 00:02:17, Tunnel0
  O     192.168.2.0/24 [110/2] via 10.0.0.12, 00:02:17, Tunnel0
  ```
  Hub2:
  ```
  O     192.168.1.0/24 [110/2] via 10.0.0.11, 00:29:15, Tunnel0
  O     192.168.2.0/24 [110/2] via 10.0.0.12, 00:29:15, Tunnel0
  ```
  Ciò significa che Hub1 e Hub2 pubblicizzeranno lo stesso costo per le reti dietro i router spoke ai router nella rete dietro i router hub. Ad esempio, la tabella di routing su un router, R2, connesso direttamente alla LAN 192.168.0.0/24, avrà il seguente aspetto:R2:
  ```
  O     IA 192.168.1.0/24 [110/12] via 192.168.0.1, 00:00:26, Ethernet1/0/3
                          [110/12] via 192.168.0.2, 00:00:27, Ethernet1/0/3O
  O     IA 192.168.2.0/24 [110/12] via 192.168.0.1, 00:00:27, Ethernet1/0/3
                          [110/12] via 192.168.0.2, 00:00:27, Ethernet1/0/3
  ```
- I router spoke hanno route a parità di costo tramite entrambi i router hub verso la rete dietro i router hub.Spoke1:
  ```
  O     IA 192.168.0.0/24 [110/11] via 10.0.0.1, 00:39:31, Tunnel0
                          [110/11] via 10.0.0.2, 00:39:31, Tunnel0
  ```
  Raggio2:
  ```
  O     IA 192.168.0.0/24 [110/11] via 10.0.0.1, 00:57:56, Tunnel0
                          [110/11] via 10.0.0.2, 00:57:56, Tunnel0
  ```
  Se i router spoke eseguono il bilanciamento del carico per pacchetto, allora i pacchetti potrebbero non essere più disponibili.

Per evitare di eseguire il routing asimmetrico o il bilanciamento del carico per pacchetto sui collegamenti ai due hub, è necessario configurare il protocollo di routing in modo che preferisca un percorso spoke-to-hub in entrambe le direzioni. Se si desidera che Hub1 sia il principale e Hub2 il backup, è possibile impostare il costo OSPF sulle interfacce del tunnel hub su un valore diverso.

Hub1:

```
interface tunnel0
...
ip ospf cost 10
...
```

Hub2:

```
interface tunnel0
...
ip ospf cost 20
...
```

Le route sono ora simili alle seguenti:

Hub1:

```
O     192.168.1.0/24 [110/11] via 10.0.0.11, 00:00:28, Tunnel0
O     192.168.2.0/24 [110/11] via 10.0.0.12, 00:00:28, Tunnel0
```
Hub2:

```
O     192.168.1.0/24 [110/21] via 10.0.0.11, 00:00:52, Tunnel0
O     192.168.2.0/24 [110/21] via 10.0.0.12, 00:00:52, Tunnel0
```
R2:

```
O     IA 192.168.1.0/24 [110/31] via 192.168.0.1, 00:01:06, Ethernet1/0/3
O     IA 192.168.2.0/24 [110/31] via 192.168.0.1, 00:01:06, Ethernet1/0/3
```
I due router hub hanno ora costi diversi sui percorsi per le reti dietro i router spoke. Ciò significa che l'hub 1 sarà preferito per l'inoltro del traffico ai router spoke, come si può notare sul router R2. In questo modo verrà risolto il problema di routing asimmetrico descritto nel primo punto precedente.

La stesura asimmetrica nella direzione opposta, descritta nel secondo punto precedente, è ancora presente. Quando si utilizza OSPF come protocollo di routing dinamico, è possibile risolvere il problema con una soluzione alternativa utilizzando il comando **distance ...** in **router ospf 1** sugli spoke per preferire le route apprese tramite Hub1 alle route apprese tramite Hub2.

Spoke1:

```
router ospf 1
 distance 111 10.0.0.2 0.0.0.0 1
access-list 1 permit any
```
Raggio2:

```
router ospf 1
 distance 111 10.0.0.2 0.0.0.0 1
access-list 1 permit any
```
Le route sono ora simili alle seguenti:

Spoke1:

```
O     192.168.0.0/24 [110/11] via 10.0.0.1, 00:00:06, Tunnel0
```
Raggio2:

```
O     192.168.1.0/24 [110/11] via 10.0.0.1, 00:00:10, Tunnel0
```
La configurazione di routing sopra riportata protegge dal routing asimmetrico, consentendo al tempo stesso il failover sull'hub 2 in caso di inattività dell'hub 1. Ciò significa che quando entrambi gli hub sono attivi, viene utilizzato solo Hub1. Se si desidera utilizzare entrambi gli hub bilanciando gli spoke tra gli hub, con protezione failover e senza routing asimmetrico, la configurazione del routing può diventare complessa, in particolare quando si utilizza OSPF. Per questo motivo, il seguente hub doppio con layout DMVPN doppio potrebbe essere una scelta migliore.

Dual Hub - Layout di DMVPN doppio

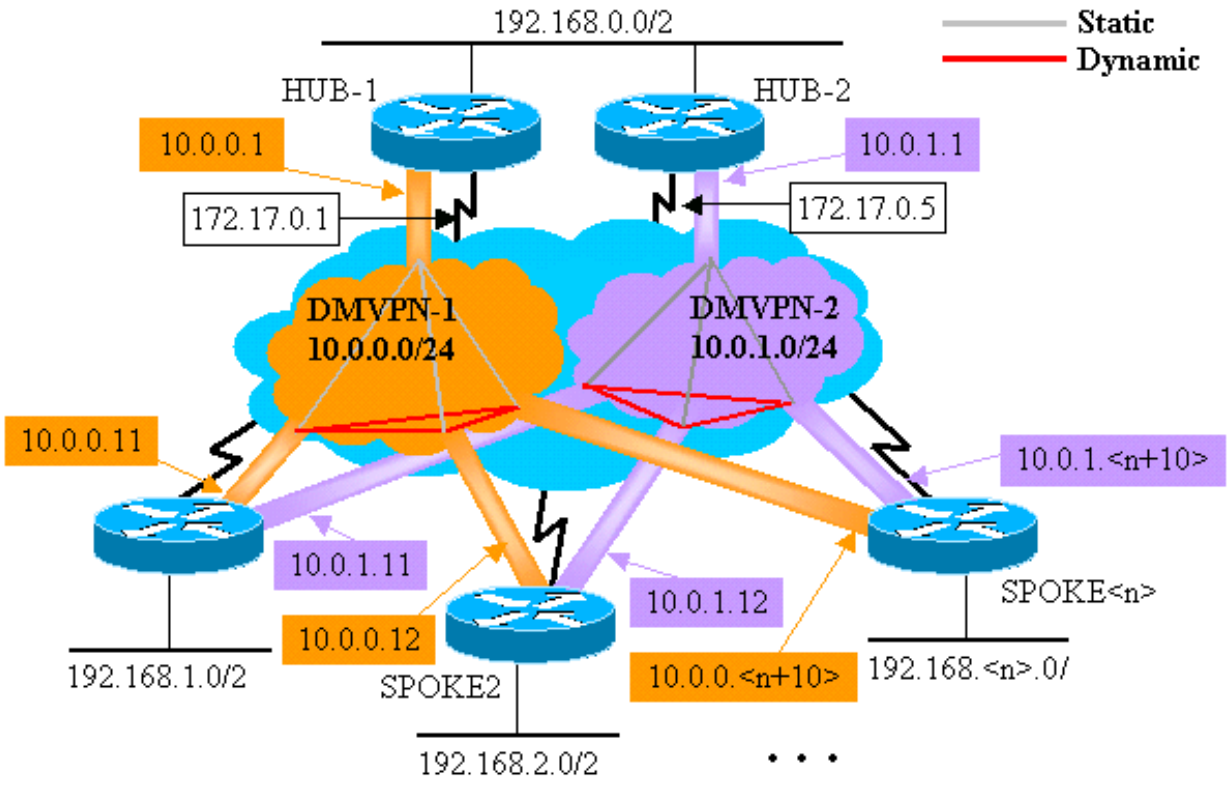
L'hub doppio con layout DMVPN doppio è leggermente più difficile da configurare, ma offre un migliore controllo del routing su DMVPN. L'idea è di avere due "cloud" DMVPN separati. Ogni hub (due in questo caso) è connesso a una subnet DMVPN ("cloud") e gli spoke sono connessi a entrambe le subnet DMVPN ("cloud"). Poiché i router spoke eseguono il routing dei router adiacenti con entrambi i router hub sulle due interfacce del tunnel GRE, è possibile utilizzare le differenze di configurazione dell'interfaccia (larghezza di banda, costo e ritardo) per modificare le metriche del protocollo di routing dinamico in modo da preferire un hub all'altro hub quando entrambi sono attivi.

Nota: questo problema in genere si verifica solo se i router dell'hub sono situati nello stesso percorso. Quando non si trovano nello stesso luogo, il normale routing dinamico finirà probabilmente per preferire il router hub corretto, anche se la rete di destinazione può essere raggiunta tramite uno dei router hub.

Èpossibile utilizzare l'interfaccia del tunnel p-GRE o mGRE sui router spoke. Più interfacce p-GRE su un router spoke possono utilizzare la stessa **origine tunnel...** Indirizzo IP, ma più interfacce GRE su un router spoke devono avere un'**origine tunnel** univoca **...** Indirizzo IP. Infatti, quando il protocollo IPsec viene avviato, il primo pacchetto è un pacchetto ISAKMP che deve essere associato a uno dei tunnel GRE. Il pacchetto ISAKMP ha solo l'indirizzo IP di destinazione (indirizzo peer IPsec remoto) con cui effettuare questa associazione. Questo indirizzo viene confrontato con l'**origine** del **tunnel...** ma poiché entrambi i tunnel hanno lo stesso indirizzo di **origine** del **tunnel...**, viene sempre confrontata la prima interfaccia del tunnel GRE. Ciò significa che i pacchetti di dati multicast in ingresso possono essere associati all'interfaccia GRE errata, interrompendo qualsiasi protocollo di routing dinamico.

I pacchetti GRE stessi non hanno questo problema perché hanno il valore della **chiave del tunnel ...** per distinguere le due interfacce GRE. A partire dal software Cisco IOS versione 12.3(5) e 12.3(7)T, è stato introdotto un parametro aggiuntivo per superare questa limitazione: **protezione tunnel....condiviso**. La parola chiave **shared** indica che più interfacce GRE utilizzeranno la crittografia IPSec con lo stesso indirizzo IP di origine. Se si dispone di una versione precedente, è possibile utilizzare i tunnel p-GRE in questo hub doppio con layout DMVPN doppio. Nel caso del tunnel p-GRE, sia l'**origine** del **tunnel ...** che la **destinazione** del **tunnel ...** Gli indirizzi IP possono essere utilizzati per la corrispondenza. Per questo esempio, in questo hub doppio con layout DMVPN doppio verranno utilizzati i tunnel p-GRE e non il qualificatore **condiviso**.

Dual Hub - Layout di DMVPN doppio



Le modifiche evidenziate riportate di seguito sono relative alle configurazioni Hub e Spoke multipoint dinamiche illustrate in precedenza in questo documento.

Router Hub1

```
version 12.3
 !
 hostname Hub1
 !
 crypto isakmp policy 1
  authentication pre-share
 crypto isakmp key cisco47 address 0.0.0.0 0.0.0.0
 !
 crypto ipsec transform-set trans2 esp-des esp-md5-hmac
  mode transport
 !
 crypto ipsec profile vpnprof
  set transform-set trans2
 !
 interface Tunnel0
  bandwidth 1000
  ip address 10.0.0.1 255.255.255.0
  ip mtu 1400
  ip nhrp authentication test
  ip nhrp map multicast dynamic
  ip nhrp network-id 100000
  ip nhrp holdtime 600
  no ip split-horizon eigrp 1
  delay 1000
  tunnel source Ethernet0
  tunnel mode gre multipoint
```

```
 tunnel key 100000
 tunnel protection ipsec profile vpnprof
!
interface Ethernet0
 ip address 172.17.0.1 255.255.255.252
!
interface Ethernet1
 ip address 192.168.0.1 255.255.255.0
!
router eigrp 1
 network 10.0.0.0 0.0.0.255
 network 192.168.0.0 0.0.0.255
 no auto-summary
!
```

Router Hub2

```
version 12.3
!
hostname Hub2
!
crypto isakmp policy 1
 authentication pre-share
crypto isakmp key cisco47 address 0.0.0.0 0.0.0.0
!
crypto ipsec transform-set trans2 esp-des esp-md5-hmac
 mode transport
!
crypto ipsec profile vpnprof
 set transform-set trans2
!
interface Tunnel0
 bandwidth 1000
 ip address 10.0.1.1 255.255.255.0
 ip mtu 1400
 ip nhrp authentication test
 ip nhrp map multicast dynamic
 ip nhrp network-id 100001
 ip nhrp holdtime 600
 no ip split-horizon eigrp 1
 delay 1000
 tunnel source Ethernet0
 tunnel mode gre multipoint
 tunnel key 100001
 tunnel protection ipsec profile vpnprof
!
interface Ethernet0
 ip address 172.17.0.5 255.255.255.252
!
interface Ethernet1
 ip address 192.168.0.2 255.255.255.0
!
router eigrp 1
 network 10.0.1.0 0.0.0.255
 network 192.168.0.0 0.0.0.255
 no auto-summary
!
```

In questo caso, le configurazioni Hub1 e Hub2 sono simili. La differenza principale consiste nel fatto che ciascuna di esse è l'hub di una DMVPN diversa. Ogni DMVPN utilizza un diverso:

- Subnet IP (10.0.0.0/24, 10.0.0.1/24)
- ID rete NHRP (100000, 100001)
- Chiave tunnel (100000, 100001)

Il protocollo di routing dinamico è stato modificato da OSPF a EIGRP, poiché è più semplice configurare e gestire una rete NBMA utilizzando EIGRP, come descritto più avanti in questo documento.

Router Spoke1

```
version 12.3
 !
 hostname Spoke1
 !
 crypto isakmp policy 1
  authentication pre-share
 crypto isakmp key cisco47 address 0.0.0.0 0.0.0.0
 !
 crypto ipsec transform-set trans2 esp-des esp-md5-hmac
  mode transport
 !
 crypto ipsec profile vpnprof
  set transform-set trans2
 !
 interface Tunnel0
  bandwidth 1000
  ip address 10.0.0.11 255.255.255.0
  ip mtu 1400
  ip nhrp authentication test
  ip nhrp map 10.0.0.1 172.17.0.1
  ip nhrp network-id 100000
  ip nhrp holdtime 300
  ip nhrp nhs 10.0.0.1
  delay 1000
  tunnel source Ethernet0
  tunnel destination 172.17.0.1
  tunnel key 100000
  tunnel protection ipsec profile vpnprof
 !
 interface Tunnel1
  bandwidth 1000
  ip address 10.0.1.11 255.255.255.0
  ip mtu 1400
  ip nhrp authentication test
  ip nhrp map 10.0.1.1 172.17.0.5
  ip nhrp network-id 100001
  ip nhrp holdtime 300
  ip nhrp nhs 10.0.1.1
  delay 1000
  tunnel source Ethernet0
  tunnel destination 172.17.0.5
  tunnel key 100001
  tunnel protection ipsec profile vpnprof
 !
 interface Ethernet0
  ip address dhcp hostname Spoke1
 !
 interface Ethernet1
  ip address 192.168.1.1 255.255.255.0
 !
 router eigrp 1
```

```
 network 10.0.0.0 0.0.0.255
 network 10.0.1.0 0.0.0.255
 network 192.168.1.0 0.0.0.255
 no auto-summary
!
```

Ognuno dei router spoke è configurato con due interfacce tunnel p-GRE, una in ciascuna delle due DMVPN. Per distinguere i due tunnel, vengono usati i valori **indirizzo ip**, **id rete ip nhrp**, **chiave tunnel e destinazione tunnel**. Il protocollo EIGRP (Dynamic Routing Protocol) viene eseguito su entrambe le subnet del tunnel p-GRE e viene utilizzato per selezionare un'interfaccia p-GRE (DMVPN) sull'altra.

Router Spoke2

```
version 12.3
!
hostname Spoke2
!
crypto isakmp policy 1
 authentication pre-share
crypto isakmp key cisco47 address 0.0.0.0 0.0.0.0
!
crypto ipsec transform-set trans2 esp-des esp-md5-hmac
 mode transport
!
crypto ipsec profile vpnprof
 set transform-set trans2
!
interface Tunnel0
 bandwidth 1000
 ip address 10.0.0.12 255.255.255.0
 ip mtu 1400
 ip nhrp authentication test
 ip nhrp map 10.0.0.1 172.17.0.1
 ip nhrp network-id 100000
 ip nhrp holdtime 300
 ip nhrp nhs 10.0.0.1
 delay 1000
 tunnel source Ethernet0
 tunnel destination 172.17.0.1
 tunnel key 100000
 tunnel protection ipsec profile vpnprof
!
interface Tunnel1
 bandwidth 1000
 ip address 10.0.1.12 255.255.255.0
 ip mtu 1400
 ip nhrp authentication test
 ip nhrp map 10.0.1.1 172.17.0.5
 ip nhrp network-id 100001
 ip nhrp holdtime 300
 ip nhrp nhs 10.0.1.1
 delay 1000
 tunnel source Ethernet0
 tunnel destination 172.17.0.5
 tunnel key 100001
 tunnel protection ipsec profile vpnprof
!
interface Ethernet0
 ip address dhcp hostname Spoke2
```

```
 !
 interface Ethernet1
  ip address 192.168.2.1 255.255.255.0
 !
 router eigrp 1
  network 10.0.0.0 0.0.0.255
  network 10.0.1.0 0.0.0.255
  network 192.168.2.0 0.0.0.255
  no auto-summary
 !
```

Router Spoke<n>

```
version 12.3
 !
 hostname Spoke<n>
 !
 crypto isakmp policy 1
  authentication pre-share
 crypto isakmp key cisco47 address 0.0.0.0 0.0.0.0
 !
 crypto ipsec transform-set trans2 esp-des esp-md5-hmac
  mode transport
 !
 crypto ipsec profile vpnprof
  set transform-set trans2
 !
 interface Tunnel0
  bandwidth 1000
  ip address 10.0.0.




  ip mtu 1400
  ip nhrp authentication test
  ip nhrp map 10.0.0.1 172.17.0.1
  ip nhrp network-id 100000
  ip nhrp holdtime 300
  ip nhrp nhs 10.0.0.1
  delay 1000
  tunnel source Ethernet0
  tunnel destination 172.17.0.1
  tunnel key 100000
  tunnel protection ipsec profile vpnprof
 !
 interface Tunnel1
  bandwidth 1000
  ip address 10.0.1.




  ip mtu 1400
  ip nhrp authentication test
  ip nhrp map 10.0.1.1 172.17.0.5
  ip nhrp network-id 100001
  ip nhrp holdtime 300
  ip nhrp nhs 10.0.1.1
  delay 1000
  tunnel source Ethernet0
```

```
 tunnel destination 172.17.0.5
 tunnel key 100001
 tunnel protection ipsec profile vpnprof
!
interface Ethernet0
 ip address dhcp hostname Spoke<x>
!
interface Ethernet1
 ip address 192.168.<n>.1 255.255.255.0
!
router eigrp 1
 network 10.0.0.0 0.0.0.255
 network 10.0.1.0 0.0.0.255
 network 192.168.<n>.0 0.0.0.255
 no auto-summary
!
```

A questo punto, diamo un'occhiata alle tabelle di routing, alle tabelle di mapping NHRP e alle connessioni IPsec sui router Hub1, Hub2, Spoke1 e Spoke2 per vedere le condizioni iniziali (subito dopo l'accensione dei router Spoke1 e Spoke2).

Condizioni iniziali e modifiche

Informazioni sul router Hub1

```
Hub1#show ip route
      172.17.0.0/30 is subnetted, 1 subnets
 C       172.17.0.0 is directly connected, Ethernet0
      10.0.0.0/24 is subnetted, 2 subnets
 C       10.0.0.0 is directly connected, Tunnel0
 D       10.0.1.0 [90/2611200] via 192.168.0.2,
00:00:46, Ethernet1
 C    192.168.0.0/24 is directly connected, Ethernet1
 D    192.168.1.0/24 [90/2841600] via 10.0.0.11,
00:00:59, Tunnel0
 D    192.168.2.0/24 [90/2841600] via 10.0.0.12,
00:00:34, Tunnel0
 Hub1#show ip nhrp
 10.0.0.12/32 via 10.0.0.12, Tunnel0 created 23:48:32,
expire 00:03:50
   Type: dynamic, Flags: authoritative unique registered
   NBMA address: 172.16.2.75
 10.0.0.11/32 via 10.0.0.11, Tunnel0 created 23:16:46,
expire 00:04:45
   Type: dynamic, Flags: authoritative unique registered
   NBMA address: 172.16.1.24
 Hub1#show crypto engine connection active
   ID Interface   IP-Address    State  Algorithm
Encrypt  Decrypt
   15 Ethernet0   172.17.63.18   set
HMAC_SHA+DES_56_CB      0        0
   16 Ethernet0   10.0.0.1       set
HMAC_SHA+DES_56_CB      0        0
 2038 Tunnel0     10.0.0.1       set
HMAC_MD5+DES_56_CB      0      759
 2039 Tunnel0     10.0.0.1       set
HMAC_MD5+DES_56_CB    726        0
 2040 Tunnel0     10.0.0.1       set
HMAC_MD5+DES_56_CB      0       37
 2041 Tunnel0     10.0.0.1       set
```

```
HMAC_MD5+DES_56_CB     36         0
```

Informazioni router Hub2

```
Hub2#show ip route
      172.17.0.0/30 is subnetted, 1 subnets
 C       172.17.0.4 is directly connected, Ethernet0
      10.0.0.0/24 is subnetted, 2 subnets
 D       10.0.0.0 [90/2611200] via 192.168.0.1,
00:12:22, Ethernet1
 C       10.0.1.0 is directly connected, Tunnel0
 C    192.168.0.0/24 is directly connected, Ethernet1
 D    192.168.1.0/24 [90/2841600] via 10.0.1.11,
00:13:24, Tunnel0
 D    192.168.2.0/24 [90/2841600] via 10.0.1.12,
00:12:11, Tunnel0
 Hub2#show ip nhrp
 10.0.1.12/32 via 10.0.1.12, Tunnel3 created 06:03:24,
expire 00:04:39
   Type: dynamic, Flags: authoritative unique registered
   NBMA address: 172.16.2.75
 10.0.1.11/32 via 10.0.1.11, Tunnel3 created 23:06:47,
expire 00:04:54
   Type: dynamic, Flags: authoritative unique registered
   NBMA address: 172.16.1.24
 Hub2#show crypto engine connection active
   ID Interface   IP-Address    State  Algorithm
Encrypt  Decrypt
    4 Ethernet0   171.17.0.5     set
HMAC_SHA+DES_56_CB      0         0
    6 Ethernet0   171.17.0.5     set
HMAC_SHA+DES_56_CB      0         0
 2098 Tunnel0     10.0.1.1       set
HMAC_MD5+DES_56_CB      0       722
 2099 Tunnel0     10.0.1.1       set
HMAC_MD5+DES_56_CB    690         0
 2100 Tunnel0     10.0.1.1       set
HMAC_MD5+DES_56_CB      0       268
 2101 Tunnel0     10.0.1.1       set
HMAC_MD5+DES_56_CB    254         0
```

Informazioni su Spoke1 Router

```
Spoke1#show ip route
      172.16.0.0/24 is subnetted, 1 subnets
 C       172.16.1.0 is directly connected, Ethernet0
      10.0.0.0/24 is subnetted, 1 subnets
 C       10.0.0.0 is directly connected, Tunnel0
 C       10.0.1.0 is directly connected, Tunnel1
 D    192.168.0.0/24 [90/2841600] via 10.0.1.1,
00:26:30, Tunnel1
                     [90/2841600] via 10.0.0.1,
00:26:30, Tunnel0
 C    192.168.1.0/24 is directly connected, Ethernet1
 D    192.168.2.0/24 [90/3097600] via 10.0.1.1,
00:26:29, Tunnel1
                     [90/3097600] via 10.0.0.1,
00:26:29, Tunnel0
 Spoke1#show ip nhrp
 10.0.0.1/32 via 10.0.0.1, Tunnel0 created 23:25:46,
never expire
   Type: static, Flags: authoritative
```

```
   NBMA address: 172.17.0.1
 10.0.1.1/32 via 10.0.1.1, Tunnel1 created 23:24:40,
never expire
   Type: static, Flags: authoritative
   NBMA address: 172.17.0.5
 Spoke1#show crypto engine connection active
   ID Interface   IP-Address    State  Algorithm
Encrypt  Decrypt
   16 Ethernet0   172.16.1.24    set
HMAC_SHA+DES_56_CB      0        0
   18 Ethernet0   172.16.1.24    set
HMAC_SHA+DES_56_CB      0        0
 2118 Tunnel0     10.0.0.11      set
HMAC_MD5+DES_56_CB      0      181
 2119 Tunnel0     10.0.0.11      set
HMAC_MD5+DES_56_CB    186        0
 2120 Tunnel1     10.0.1.11      set
HMAC_MD5+DES_56_CB      0      105
 2121 Tunnel1     10.0.1.11      set
HMAC_MD5+DES_56_CB    110        0
```

Informazioni router Spoke2

```
Spoke2#show ip route
      172.16.0.0/24 is subnetted, 1 subnets
 C       172.16.2.0 is directly connected, Ethernet0
      10.0.0.0/24 is subnetted, 2 subnets
 C       10.0.0.0 is directly connected, Tunnel0
 C       10.0.1.0 is directly connected, Tunnel1
 D    192.168.0.0/24 [90/2841600] via 10.0.1.1,
00:38:04, Tunnel1
                     [90/2841600] via 10.0.0.1,
00:38:04, Tunnel0
 D    192.168.1.0/24 [90/3097600] via 10.0.1.1,
00:38:02, Tunnel1
                     [90/3097600] via 10.0.0.1,
00:38:02, Tunnel0
 C    192.168.2.0/24 is directly connected, Ethernet1
 Spoke2#show ip nhrp
 10.0.0.1/32 via 10.0.0.1, Tunnel0 created 1d02h, never
expire
   Type: static, Flags: authoritative used
   NBMA address: 172.17.0.1
 10.0.1.1/32 via 10.0.1.1, Tunnel1 created 1d02h, never
expire
   Type: static, Flags: authoritative used
   NBMA address: 172.17.0.5
 Spoke2#show crypto engine connection active
   ID Interface   IP-Address    State  Algorithm
Encrypt  Decrypt
    8 Ethernet0   172.16.2.75    set
HMAC_SHA+DES_56_CB      0        0
    9 Ethernet0   172.16.2.75    set
HMAC_SHA+DES_56_CB      0        0
 2036 Tunnel0     10.0.0.12      set
HMAC_MD5+DES_56_CB      0      585
 2037 Tunnel0     10.0.0.12      set
HMAC_MD5+DES_56_CB    614        0
 2038 Tunnel1     10.0.1.12      set
HMAC_MD5+DES_56_CB      0      408
 2039 Tunnel1     10.0.1.12      set
HMAC_MD5+DES_56_CB    424        0
```

Di nuovo, è possibile notare un paio di aspetti interessanti sulle tabelle di routing su Hub1, Hub2, Spoke1 e Spoke2:

- Entrambi i router hub hanno route a parità di costo verso le reti dietro i router spoke.Hub1:
  ```
  D    192.168.1.0/24 [90/2841600] via 10.0.0.11, 00:00:59, Tunnel0
  D    192.168.2.0/24 [90/2841600] via 10.0.0.12, 00:00:34, Tunnel0
  ```
  Hub2:
  ```
  D    192.168.1.0/24 [90/2841600] via 10.0.1.11, 00:13:24, Tunnel0
  D    192.168.2.0/24 [90/2841600] via 10.0.1.12, 00:12:11, Tunnel0
  ```
  Ciò significa che Hub1 e Hub2 pubblicizzeranno lo stesso costo per le reti dietro i router spoke ai router nella rete dietro i router hub. Ad esempio, la tabella di routing su un router, R2, connesso direttamente alla LAN 192.168.0.0/24, avrà il seguente aspetto:R2:
  ```
  D    192.168.1.0/24 [90/2867200] via 192.168.0.1, 00:51:51, Ethernet1/0/3
                              [90/2867200] via 192.168.0.2, 00:51:51, Ethernet1/0/3
  D    192.168.2.0/24 [90/2867200] via 192.168.0.2, 00:52:43, Ethernet1/0/3
                      [90/2867200] via 192.168.0.1, 00:52:43, Ethernet1/0/3
  ```
- I router spoke hanno route a parità di costo tramite entrambi i router hub verso la rete dietro i router hub.Spoke1:
  ```
  D    192.168.0.0/24 [90/3097600] via 10.0.1.1, 00:26:30, Tunnel1
                              [90/3097600] via 10.0.0.1, 00:26:30, Tunnel0
  ```
  Raggio2:
  ```
  D    192.168.0.0/24 [90/3097600] via 10.0.1.1, 00:38:04, Tunnel1
                              [90/3097600] via 10.0.0.1, 00:38:04, Tunnel0
  ```
  Se i router spoke eseguono il bilanciamento del carico per pacchetto, allora i pacchetti potrebbero non essere più disponibili.

Per evitare di eseguire il routing asimmetrico o il bilanciamento del carico per pacchetto sui collegamenti ai due hub, è necessario configurare il protocollo di routing in modo che preferisca un percorso spoke-to-hub in entrambe le direzioni. Se si desidera che Hub1 sia il principale e Hub2 il backup, è possibile impostare un ritardo diverso per le interfacce del tunnel hub.

Hub1:

```
interface tunnel0
...
delay 1000
...
```

Hub2:

```
interface tunnel0
...
delay 1050
...
```

Nota: nell'esempio, è stato aggiunto 50 al ritardo sull'interfaccia del tunnel sull'hub 2 perché è inferiore al ritardo sull'interfaccia Ethernet 1 tra i due hub (100). In questo modo, l'hub 2 continua a inoltrare i pacchetti direttamente ai router spoke, ma annuncia un percorso meno desiderabile rispetto all'hub 1 ai router dietro l'hub 1 e l'hub 2. Se il ritardo è stato aumentato di oltre 100, l'hub 2 inoltra i pacchetti per i router spoke attraverso l'hub 1 tramite l'interfaccia Ethernet1, sebbene i router dietro l'hub 1 e l'hub 2 preferiscano ancora correttamente l'hub 1 per inviare i pacchetti ai router spoke.

Le route sono ora simili alle seguenti:

Hub1:

```
D    192.168.1.0/24 [90/2841600] via 10.0.0.11, 00:01:11, Tunnel0
D    192.168.2.0/24 [90/2841600] via 10.0.0.12, 00:01:11, Tunnel0
```

Hub2:

```
D    192.168.1.0/24 [90/2854400] via 10.0.1.11, 00:00:04, Tunnel0
D    192.168.2.0/24 [90/2854400] via 10.0.1.12, 00:00:04, Tunnel0
```

R2:

```
D    192.168.1.0/24 [90/2867200] via 192.168.0.1, 00:02:18, Ethernet1/0/3
D    192.168.2.0/24 [90/2867200] via 192.168.0.1, 00:02:18, Ethernet1/0/3
```

I due router hub hanno costi diversi per le route di rete dietro i router spoke, quindi, in questo caso, Hub1 sarà preferito per l'inoltro del traffico ai router spoke, come si può vedere in R2. Questo risolve il problema descritto nel primo punto precedente.

Il problema descritto nel secondo punto precedente è ancora presente, ma poiché si dispone di due interfacce tunnel p-GRE, è possibile impostare il **ritardo ...** sulle interfacce tunnel separatamente per modificare la metrica EIGRP per le route apprese dall'hub1 rispetto all'hub2.

Spoke1:

```
interface tunnel0
  delay 1000
interface tunnel1
  delay 1050
```

Raggio2:

```
interface tunnel0
  delay 1000
interface tunnel1
  delay 1050
```

Le route sono ora simili alle seguenti:

Spoke1:

```
D    192.168.0.0/24 [90/2841600] via 10.0.0.1, 00:15:44, Tunnel0
D    192.168.2.0/24 [90/3097600] via 10.0.0.1, 00:15:44, Tunnel0
```

Raggio2:

```
D    192.168.0.0/24 [90/2841600] via 10.0.0.1, 00:13:54, Tunnel0
D    192.168.1.0/24 [90/3097600] via 10.0.0.1, 00:13:54, Tunnel0
```

La configurazione di routing sopra riportata protegge dal routing asimmetrico, consentendo al tempo stesso il failover sull'hub 2 in caso di inattività dell'hub 1. Ciò significa che quando entrambi gli hub sono attivi, viene utilizzato solo Hub1.

Se si desidera utilizzare entrambi gli hub eseguendo il bilanciamento degli spoke tra gli hub, con protezione failover e senza routing asimmetrico, la configurazione del routing è più complessa, ma è possibile farlo con il protocollo EIGRP. A tale scopo, impostare il **ritardo ...** sulle interfacce tunnel dei router dell'hub su Uguale e quindi utilizzare il comando **offset-list <acl> out <offset>**

<interface> sui router spoke per aumentare la metrica EIGRP per i percorsi annunciati dalle interfacce del tunnel GRE all'hub di backup. Il **ritardo** diseguale **...** tra le interfacce Tunnel0 e Tunnel1 sullo spoke è ancora utilizzato, quindi il router spoke preferirà il proprio router hub primario. Le modifiche sui router spoke sono le seguenti.

Router Spoke1

```
version 12.3
 !
 hostname Spoke1
 !
 ...
 !
 interface Tunnel0
  bandwidth 1000
  ip address 10.0.0.11 255.255.255.0
  ip mtu 1400
  ip nhrp authentication test
  ip nhrp map 10.0.0.1 172.17.0.1
  ip nhrp network-id 100000
  ip nhrp holdtime 300
  ip nhrp nhs 10.0.0.1
  delay 1000
  tunnel source Ethernet0
  tunnel destination 172.17.0.1
  tunnel key 100000
  tunnel protection ipsec profile vpnprof
 !
 interface Tunnel1
  bandwidth 1000
  ip address 10.0.1.11 255.255.255.0
  ip mtu 1400
  ip nhrp authentication test
  ip nhrp map 10.0.1.1 172.17.0.5
  ip nhrp network-id 100001
  ip nhrp holdtime 300
  ip nhrp nhs 10.0.1.1
  delay 1500
  tunnel source Ethernet0
  tunnel destination 172.17.0.5
  tunnel key 100001
  tunnel protection ipsec profile vpnprof
 !
 ...
 !
 router eigrp 1
  offset-list 1 out 12800 Tunnel1
  network 10.0.0.0 0.0.0.255
  network 10.0.1.0 0.0.0.255
  network 192.168.1.0
  distribute-list 1 out
  no auto-summary
 !
 access-list 1 permit 192.168.1.0
 !
```

Router Spoke2

```
version 12.3
 !
```

```
hostname Spoke2
!
...
!
interface Tunnel0
 bandwidth 1000
 ip address 10.0.0.12 255.255.255.0
 ip mtu 1400
 ip nhrp authentication test
 ip nhrp map 10.0.0.1 172.17.0.1
 ip nhrp network-id 100000
 ip nhrp holdtime 300
 ip nhrp nhs 10.0.0.1
 delay 1500
 tunnel source Ethernet0
 tunnel destination 172.17.0.1
 tunnel key 100000
 tunnel protection ipsec profile vpnprof
!
interface Tunnel1
 bandwidth 1000
 ip address 10.0.1.12 255.255.255.0
 ip mtu 1400
 ip nhrp authentication test
 ip nhrp map 10.0.1.1 172.17.0.5
 ip nhrp network-id 100001
 ip nhrp holdtime 300
 ip nhrp nhs 10.0.1.1
 delay 1000
 tunnel source Ethernet0
 tunnel destination 172.17.0.5
 tunnel key 100001
 tunnel protection ipsec profile vpnprof
!
...
!
router eigrp 1
 offset-list 1 out 12800 Tunnel1
 network 10.0.0.0 0.0.0.255
 network 10.0.1.0 0.0.0.255
 network 192.168.2.0
 distribute-list 1 out
 no auto-summary
!
access-list 1 permit 192.168.2.0
!
```

Nota: il valore di offset di 12800 (50*256) è stato aggiunto alla metrica EIGRP perché è inferiore a 25600 (100*256). Questo valore (25600) viene aggiunto alla metrica EIGRP per le route apprese tra i router hub. Utilizzando il comando **offset-list** del numero 12800, il router dell'hub di backup inoltra i pacchetti direttamente ai router spoke, anziché inoltrarli tramite Ethernet per passare attraverso il router dell'hub primario per tali spoke. La metrica sulle route annunciate dai router hub sarà ancora tale da preferire il router hub primario corretto. Tenere presente che la metà degli spoke hanno Hub1 come router principale, mentre l'altra metà hanno Hub2 come router principale.

Nota: se il valore di offset è stato aumentato di oltre 25600 (100*256), gli hub inoltrerebbero i pacchetti per metà dei router spoke attraverso l'altro hub tramite l'interfaccia Ethernet1, anche se i router dietro gli hub preferirebbero comunque l'hub corretto per inviare i pacchetti ai router spoke.

Nota: è stato aggiunto anche il comando **distribute-list 1 out** perché è possibile che le route apprese da un router hub tramite un'interfaccia tunnel su uno spoke vengano annunciate all'altro

hub tramite l'altro tunnel. Il comando **distribute-list ...** assicura che il router spoke possa annunciare solo le proprie route.

Nota: se si preferisce controllare gli annunci di routing sui router hub anziché sui router spoke, è possibile configurare l'**offset-list <acl1> in <value> <interface>** e **distribute-list <acl2>** nei comandi sui router hub anziché sugli spoke. Nell'elenco <acl2> access-list vengono elencate le route da dietro tutti i spoke, mentre nell'elenco <acl1> access-list vengono elencate solo le route da dietro i spoke dove un altro router hub deve essere l'hub primario.

Con queste modifiche le route sono simili alle seguenti:

Hub1:

```
D    192.168.1.0/24 [90/2841600] via 10.0.0.11, 00:12:11, Tunnel2
D    192.168.2.0/24 [90/2854400] via 10.0.0.12, 00:13:24, Tunnel2
```

Hub2:

```
D    192.168.1.0/24 [90/2854400] via 10.0.1.11, 00:09:58, Tunnel0
D    192.168.2.0/24 [90/2841600] via 10.10.1.12, 00:11:11, Tunnel0
```

R2:

```
D    192.168.1.0/24 [90/2867200] via 192.168.0.1, 00:13:13, Ethernet1/0/3
D    192.168.2.0/24 [90/2867200] via 192.168.0.2, 00:14:25, Ethernet1/0/3
```

Spoke1:

```
D    192.168.0.0/24 [90/2841600] via 10.0.0.1, 00:16:12, Tunnel0
```

Raggio2:

```
D    192.168.0.0/24 [90/2841600] via 10.0.1.1, 00:18:54, Tunnel1
```

Conclusioni

La soluzione DMVPN fornisce le seguenti funzionalità per migliorare la scalabilità delle reti VPN IPsec di grandi e piccole dimensioni.

- DMVPN consente una migliore scalabilità in VPN IPsec a rete completa o parziale.Èparticolarmente utile quando il traffico spoke-to-spoke è sporadico (ad esempio, ogni spoke non invia costantemente dati a tutti gli altri spoke). Consente a qualsiasi spoke di inviare dati direttamente a qualsiasi altro spoke, a condizione che esista una connettività IP diretta tra gli spoke.
- DMVPN supporta i nodi IPsec con indirizzi assegnati in modo dinamico, ad esempio Cable, ISDN e DSL.Ciò si applica alle reti hub e spoke e mesh. DMVPN può richiedere che il collegamento hub-to-spoke sia sempre attivo.
- DMVPN semplifica l'aggiunta di nodi VPN.Quando si aggiunge un nuovo router spoke, è sufficiente configurare il router spoke e collegarlo alla rete (tuttavia, potrebbe essere necessario aggiungere informazioni di autorizzazione ISAKMP per il nuovo spoke sull'hub). L'hub acquisirà informazioni dinamiche sul nuovo spoke e il protocollo di routing dinamico

propagherà il routing all'hub e a tutti gli altri spoke.
- DMVPN riduce le dimensioni della configurazione necessaria su tutti i router della VPN.Questo è anche il caso delle reti VPN hub e spoke GRE+IPsec.
- DMVPN utilizza il GRE e, pertanto, supporta il multicast IP e il traffico di routing dinamico attraverso la VPN.Ciò significa che è possibile utilizzare un protocollo di routing dinamico e che il protocollo può supportare "hub" ridondanti. Sono supportate anche le applicazioni multicast.
- DMVPN supporta il tunneling suddiviso negli spoke.

Informazioni correlate

- DMVPN (Dynamic Multipoint VPN)
- Pagina di supporto per IPSec
- Supporto tecnico – Cisco Systems