动态多点 IPsec VPN（使用多点 GRE/NHRP 扩大 IPsec VPN）

目录

简介

本文档讨论了动态多点 IPsec VPN (DMVPN) 以及公司为何需要设计或迁移其网络以便在 Cisco IOS® 软件中使用此新的 IPsec VPN 解决方案。

背景信息

公司可能需要在加密数据流以对其进行保护的同时，通过 Internet 使多个站点与一个主站点互连，或许还需要使这些站点彼此互连。例如，有一组零售商店需要连接到公司总部以获得库存和进行订购，这些商店可能还需要连接到公司的其他商店以核实产品的可用性。以前，建立连接的唯一方法是使用第 2 层网络（如 ISDN 或帧中继）实现所有互连。这些硬连接链路的内部 IP 数据流设置和付费既费时又昂贵。如果所有站点（包括主站点）已具有成本相对较低的 Internet 访问，则此 Internet 访问还可用于商店与总部之间的内部 IP 通信，并通过使用 IPsec 隧道来确保保密性和数据完整性。

为使公司建立实现其站点跨 Internet 互连的大型 IPsec 网络，您必须能够调整 IPsec 网络的规模。IPsec 对两个端点（对等体）之间的数据流进行加密，加密由两个端点使用共享“密钥”完成。由于此密钥仅在这两个端点之间共享，因此加密网络本质上是点对点链路的集合。所以从本质上讲，IPsec 是点对点隧道网络。调整大型点对点网络的最可行的方法是将其组织为星型网络或完全（部分）网状网络。在大多数网络中，IP 数据流的大部分在分支与中心之间，只有极少部分在分支之间，因此星型设计通常是最佳选择。这种设计也适用于早期的帧中继网络，因为要为这些网络中所有站点之间的链路付费过于昂贵。

使用 Internet 作为中心与分支之间的互连方式时，分支彼此之间也可进行直接访问而不需要附加成本，但要建立并/或管理完全（部分）网状网络，即使可能实现也是非常困难的。通常需要使用完全或部分网状网络的原因在于，如果分支到分支的数据流能够直接通过中心而不是经由中心传输，便可以节省成本。通过中心的分支到分支数据流会使用中心资源且可能导致额外延迟（特别是采用 IPsec 加密时），因为中心需要对来自发送分支的传入数据包进行解密，再对数据流进行重新加密以将其发送到接收分支。还有一个适合使用直接分支到分支数据流的示例，即两个分支位于同一个城市而中心位于国家/地区中其他地点的情况。

随着 IPsec 星型网络的部署和壮大，越来越需要使它们尽可能以动态方式路由 IP 数据包。在早期的帧中继星型网络中，是通过在帧中继链路上运行动态路由协议（如 OSPF 或 EIGRP）实现这一点的。这对于动态广播分支网络的可达性以及支持 IP 路由网络的冗余很有用处。如果网络的中心路由器无法工作，备用中心路由器可自动接管以保持与分支网络的网络连接。

IPsec 隧道和动态路由协议有一个基本问题。动态路由协议需要使用 IP 多播或广播数据包，但 IPsec 不支持多播或广播数据包的加密。目前解决此问题的方法是将通用路由封装 (GRE) 隧道与 IPsec 加密结合使用。

GRE 隧道支持将 IP 多播和广播数据包传输到 GRE 隧道的另一端。GRE 隧道数据包是 IP 单播数据包，因此可以使用 IPsec 对 GRE 数据包进行加密。在这种情况下，由 GRE 完成隧道工作，由 IPsec 完成支持 VPN 网络的加密部分。配置 GRE 隧道时，另一个端点必须了解隧道端点的 IP 地址（**tunnel source ...**、**tunnel destination ...**），而且这些地址必须可以通过 Internet 进行路由。这意味着此网络中的中心路由器和所有分支路由器必须具有静态非专用 IP 地址。

对于到 Internet 的小型站点连接，分支的外部 IP 地址通常会在每次连接到 Internet 时发生更改，因为它们的 Internet 服务提供商 (ISP) 会在每次分支上线（非对称数字用户线 (ADSL) 和电缆服务）时（通过动态主机配置协议 (DHCP)）动态提供外部接口地址。动态分配路由器的“外部地址”可使 ISP 超额预定其 Internet 地址空间的使用状况，因为所有用户并不会同时联机。如果向提供商支付为分支路由器分配静态地址的费用，可能会昂贵得多。通过 IPsec VPN 运行动态路由协议需要使用 GRE 隧道，但是您将无法使用其外部物理接口具有动态分配 IP 地址的分支。

以下四点总结了上述限制及一些其他限制：

- IPsec 使用访问控制列表 (ACL) 定义要对哪些数据进行加密。因此，每次在分支或中心后面添加新的（子）网络时，用户都必须更改中心路由器和分支路由器上的 ACL。如果由 SP 管理路由器，用户必须通知 SP 以更改 IPsec ACL，使新数据流能够获得加密。
- 使用大型星型网络时，中心路由器配置的大小可能变得非常大，甚至达到使其不可用的程度。

例如，要支持 300 个分支路由器，中心路由器需要多达 3900 行的配置。这种大规模配置使得显示配置和查找与当前所调试问题相关的配置部分变得十分困难。此外，这种大小的配置过大，NVRAM 无法容纳，需要存储在闪存中。

- GRE 和 IPsec 必须了解端点对等体地址。分支的 IP 地址通过其自己的 ISP 直接连接到 Internet，而且它们通常设置为外部接口地址不固定。每次站点（通过 DHCP）联机时，IP 地址都会更改。
- 如果分支需要通过 IPsec VPN 与彼此直接通信，那么星型网络必须变为完全网状网络。由于尚未确定哪些分支需要与彼此直接通信，因此尽管可能并非每个分支都需要与所有其他分支直接通信，仍然需要完全网状网络。而且，在小型分支路由器上配置 IPsec 以使其与网络中所有其他分支路由器直接连接也是不可行的；因此分支路由器需要具有更强大的功能。

DMVPN 解决方案

DMVPN 解决方案使用多点 GRE (mGRE) 和下一跳解析协议 (NHRP)，并具有 IPsec 和一些新的增强功能，用于以可调节的方式解决上述问题。

IPsec 加密自动启动

如果未使用 DMVPN 解决方案，则在出现需要使用 IPsec 加密隧道的数据流之前，不会建立该 IPsec 隧道。完成 IPsec 隧道的建立可能需要 1 到 10 秒的时间，在此期间会丢弃数据流。将 GRE 与 IPsec 一起使用时，GRE 隧道配置已包括 GRE 隧道对等体 (**tunnel destination …**)地址，它也是 IPsec 对等体地址。这两个地址都是预先配置的。

如果在中心路由器上使用隧道端点发现 (TED) 和动态加密映射，则无需在中心预先配置 IPsec 对等体地址，但需要先发送并接收 TED 探测器和响应，才能开始 ISAKMP 协商。这应当不是必需的，因为使用 GRE 时，对等体源地址和目标地址是已知的。它们要么在配置中，要么使用 NHRP 进行解析（多点 GRE 隧道）。

使用 DMVPN 解决方案，点对点和多点 GRE 隧道都会立即触发 IPsec。而且不需要配置任何加密 ACL，因为它们将自动从 GRE 隧道源地址和目标地址生成。以下命令用于定义 IPsec 加密参数。请注意，不需要任何 **set peer ...** 或 **match address ...** 命令，因为此信息直接从关联的 GRE 隧道或 NHRP 映射生成。

```
crypto ipsec profile <profile-name> set transform-set <transform-name>
```

以下命令将一个隧道接口与 IPsec 配置文件关联起来。

```
interface tunnel<number> ... tunnel protection ipsec profile <profile-name>
```

“分支到中心”链路的动态隧道创建

在 DMVPN 网络中，中心路由器上不会配置有关某个分支的任何 GRE 或 IPsec 信息。分支路由器的 GRE 隧道使用有关中心路由器的信息（通过 NHRP 命令）进行配置。当分支路由器启动时，它会自动建立连接中心路由器的 IPsec 隧道，如上所述。随后使用 NHRP 将其当前物理接口 IP 地址告知中心路由器。这非常实用，原因有以下三个：

- 如果分支路由器的物理接口 IP 地址是动态分配的（例如使用 ADSL 或 CableModem），则无法使用此信息配置中心路由器，因为每次分支路由器重新加载时都会获得新的物理接口 IP 地址。
- 由于不需要有任何有关对等路由器的 GRE 或 IPsec 信息，因此可以缩短并简化中心路由器的

配置。所有这些信息都可以通过 NHRP 动态学习。

- 为 DMVPN 网络添加新的分支路由器时，不需要更改中心或任何当前分支路由器上的配置。新的分支路由器使用中心路由器的信息进行配置，当它启动时，会在中心路由器进行动态注册。动态路由协议将此分支的路由信息传播到中心路由器。中心路由器再将这些新路由信息传播到其他分支。此外，它也会将来自其他分支的路由信息传播到此分支。

“分支到分支”数据流的动态隧道创建

如前所述，目前在网状网络中，必须在所有路由器上配置所有点对点 IPsec（或 IPsec+GRE）隧道，即使其中一些或大多数隧道并未运行或并非一直需要这些隧道也必须如此。使用 DMVPN 解决方案则是将一个路由器作为中心路由器，所有其他路由器（分支）配置连接该中心的隧道。分支到中心的隧道始终打开，分支不需要配置到任何其他分支的直接隧道。当某个分支需要将数据包传输至另一分支（如另一分支后面的子网）时，会使用 NHRP 动态确定目标分支所需的目标地址。中心路由器充当 NHRP 服务器，并处理源分支的这一请求。随后两个分支（通过同一 mGRE 接口）以动态方式创建他们之间的 IPsec 隧道，这样便可以直接传输数据了。经过一个可配置的非活动周期之后，将自动切断这个分支到分支的动态隧道。

支持动态路由协议

DMVPN 解决方案以支持隧道组播/广播 IP 数据包的 GRE 隧道为基础，因此 DMVPN 解决方案也支持通过 IPsec+mGRE 隧道运行的动态路由协议。以前，NHRP 需要用户明确配置广播/组播映射的隧道目标 IP 地址，才能支持组播和广播 IP 数据包 GRE 隧道。例如，在中心路由器上，需要为每个分支设置 **ip nhrp map multicast <spoke-n-addr>** 配置行。使用 DMVPN 解决方案时，并未预先了解分支地址，因此不可能使用这种配置。可以将 NHRP 配置为自动使用 **ip nhrp map multicast dynamic** 命令将每个分支添加到中心路由器的组播目标列表。使用此命令，当分支路由器在 NHRP 服务器（中心路由器）注册它们的单播 NHRP 映射时，NHRP 也会为此分支创建广播/组播映射。这样，就不需要预先了解分支地址了。

对 mGRE 的 Cisco 快速转发快速交换

目前 mGRE 接口的数据流采用进程交换方式，因此导致性能较低。DMVPN 解决方案添加了用于 mGRE 数据流的 Cisco 快速转发交换，从而大大提高了性能。启用此功能无需使用任何配置命令。如果 GRE 隧道接口和传出/传入物理接口允许使用 Cisco 快速转发交换，那么多点 GRE 隧道数据包将采用 Cisco 快速转发交换方式。

在 IPsec 保护的 VPN 上使用动态路由

本部分介绍当前的（使用 DMVPN 解决方案之前的）情况。在 Cisco 路由器上实现 IPsec 的方法是使用一组定义加密的命令，然后在路由器外部接口应用 **crypto map <map-name>** 命令。由于是这种设计，而且目前没有使用 IPsec 加密 IP 多播/广播数据包的标准，因此无法通过 IPsec 隧道“转发”IP 路由协议数据包，也不能将任何路由更改动态传播到 IPsec 隧道的另一端。

注意： 除 BGP 外，所有动态路由协议均使用广播或组播 IP 数据包。将 GRE 隧道与 IPsec 结合使用可解决此问题。

在 Cisco 路由器上实现 GRE 隧道的方法是使用虚拟隧道接口 (**interface tunnel<#>**)。GRE 隧道协议设计用于处理 IP 多播/广播数据包，因此动态路由协议可以“越过”GRE 隧道。GRE 隧道数据包是封装原始 IP 多播/单播数据包的 IP 单播数据包。您随后可以使用 IPsec 对 GRE 隧道数据包进行加密。此外，还可以在传输模式下运行 IPsec 并节省 20 个字节，因为 GRE 已封装原始数据包，所以不需要 IPsec 在另一 IP 报头中封装 GRE IP 数据包。

在传输模式下运行 IPsec 时，要加密的数据包的 IP 源地址和目标地址必须与 IPsec 对等体地址（路由器本身）匹配。在这种情况下，这意味着 GRE 隧道端点与 IPsec 对等体地址必须相同。由于路由器同时作为 IPsec 和 GRE 隧道端点，因此这不是问题。通过结合 GRE 隧道与 IPsec 加密，可以使用动态 IP 路由协议更新加密隧道两端的路由表。通过加密隧道学习的网络 IP 路由表条目会将隧道另一端（GRE 隧道接口 IP 地址）作为 IP 下一跳。因此，如果隧道任一端的网络发生更改，另一端将动态学习该更改，连接也将得以保持，而无需在路由器上进行任何配置更改。

基本配置

下面是标准点对点 IPsec+GRE 配置。此配置后面是一系列配置示例，这些示例逐步添加 DMVPN 解决方案的特定功能以显示 DMVPN 的不同功能。每个示例都以前面的示例为基础，以显示如何在复杂程度逐渐增加的网络设计中使用 DMVPN 解决方案。这一系列示例可用作将当前 IPsec+GRE VPN 迁移到 DMVPN 的模板。如果该特定配置示例符合您的网络设计要求，您可以随时停止“迁移”。

IPsec + GRE 星型网络 (n = 1,2,3,...)

中心路由器

```
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.<n>.1 set
transform-set trans2 match address <n+100> ! 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<n> bandwidth 1000 ip address
10.0.0.<4n-3> 255.255.255.252 ip mtu 1400 delay 1000
tunnel source Ethernet0 tunnel destination 172.16.<n>.1
! 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 <n+100>
permit gre host 172.17.0.1 host 172.16.<n>.1
```

Spoke1 路由器

```
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
```

Spoke2 路由器

```
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
```

Spoke<n> 路由器

```
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.<n>.0 0.0.0.255 no
auto-summary ! access-list 101 permit gre host
172.16.<n>.1 host 172.17.0.1
```

在上述配置中，使用 ACL 定义将对哪些数据流进行加密。在中心路由器和分支路由器上，此 ACL 只需要与 GRE 隧道 IP 数据包匹配。无论任一端的网络如何更改，GRE IP 隧道数据包都不会更改，所以此 ACL 也不需要更改。

注意： 使用 12.2(13)T 之前的 Cisco IOS 软件版本时，必须对 GRE 隧道接口 (Tunnel<x>) 和物理接口 (Ethernet0) 应用 **crypto map vpnmap1 configuration** 命令。使用 Cisco IOS 版本 12.2(13)T 和更高版本时，只需对物理接口 (Ethernet0) 应用 **crypto map vpnmap1 configuration** 命令。

中心和分支路由器的路由表示例

中心路由器上的路由表

```
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>
```

Spoke1 路由器上的路由表

```
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
```

Spoke<n> 路由器上的路由表

```
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
```

这是一种基本工作配置，用作与使用 DMVPN 解决方案时可能的更复杂配置进行比较的起点。第一项更改将减少中心路由器上配置的大小。这对于数量较小的分支路由器并不重要，但当分支路由器达到 50 到 100 个时，就变得非常关键。

减小中心路由器配置的大小

在下面的示例中，中心路由器上的配置以最小更改幅度从多个 GRE 点对点隧道接口更改为单个 GRE 多点隧道接口。这是迁移到 DMVPN 解决方案的第一步。

中心路由器上有一段唯一的配置行，用于定义每个分支路由器的加密映射特征。这部分配置定义了该分支路由器的加密 ACL 和 GRE 隧道接口。除 IP 地址（**set peer …**、**tunnel destination …**）外，所有分支的这些特征几乎都相同。

观察中心路由器的上述配置，会看到每个分支路由器至少有 13 行配置；其中四行用于加密映射，一行用于加密 ACL，八行用于 GRE 隧道接口。如果有 300 个分支路由器，配置行的总数就是 3900。此外，每个隧道链路的寻址还需要 300 (/30) 个子网。这种大小的配置非常难以管理，对 VPN 网络进行故障排除时甚至会更加困难。要降低此值，可以使用动态加密映射，这样会使上述值减少 1200 行，即 300 个分支网络中有 2700 行配置。

注意： 使用动态加密映射时，必须由分支路由器建立 IPsec 加密隧道。此外，也可以使用 **ip unnumbered <interface>** 减少 GRE 隧道所需的子网数量，但这可能会使以后进行故障排除的难度增加。

使用 DMVPN 解决方案，可以在中心路由器上配置单个多点 GRE 隧道接口和单个 IPsec 配置文件，用于处理所有分支路由器。这样，无论在 VPN 网络中添加多少分支路由器，中心路由器上配置的大小都可以保持在某个常量。

DMVPN 解决方案引入了以下新命令：

```
crypto ipsec profile <name> <ipsec parameters> tunnel protection ipsec profile <name> ip nhrp map multicast dynamic
```

crypto ipsec profile <name> 命令的用法类似动态加密映射，它是专门针对隧道接口设计的。此命令用于定义分支到中心和分支到分支 VPN 隧道的 IPsec 加密的参数。此配置文件下所需的唯一参数是转换集。IPsec 对等体地址和 IPsec 代理的 **match address ...** 子句是自动从 GRE 隧道的 NHRP 映射生成的。

tunnel protection ipsec profile <name> 命令在 GRE 隧道接口下配置，用于将 GRE 隧道接口与 IPsec 配置文件关联起来。另外，**tunnel protection ipsec profile <name>** 命令还可与点对点 GRE 隧道一起使用。在这种情况下，它将从 **tunnel source ...** 和 **tunnel destination ...** 配置中生成 IPsec 对等体和代理的信息。这样可以简化配置，因为不再需要 IPsec 对等体和加密 ACL。

注意： **tunnel protection …** 命令指定 IPsec 加密将在 GRE 封装添加到数据包中后完成。

这前两个新命令类似于使用 **crypto map <name>** 命令配置加密映射和将加密映射分配到接口。最大的区别在于，使用新命令时不需要指定与要加密的数据包匹配的 IPsec 对等体地址或 ACL。系统将根据 mGRE 隧道接口的 NHRP 映射自动确定这些参数。

注意： 在隧道接口使用 **tunnel protection …** 命令时，不会在物理传出接口上配置 **crypto map ...** 命

令。

最后一个新命令 **ip nhrp map multicast dynamic** 在分支路由器建立 mGRE+IPsec 隧道并注册其单播 NHRP 映射时允许 NHRP 将这些分支路由器添加到组播 NHRP 映射中。启用跨中心和分支之间的 mGRE+IPsec 隧道工作的动态路由协议时，需要使用此命令。如果此命令不可用，中心路由器就需要针对到每个分支的组播映射设置一个单独的配置行。

注意： 使用此配置时，必须由分支路由器建立 mGRE+IPsec 隧道连接，因为中心路由器没有配置任何与分支有关的信息。但这并不是问题，因为使用 DMVPN 解决方案，当分支路由器启动时会自动建立 mGRE+IPsec 隧道，而且该隧道始终保持打开状态。

注意： 下面的示例显示分支路由器上的点对点 GRE 隧道接口，以及在中心路由器和分支路由器上添加的用于支持中心路由器上 mGRE 隧道的 NHRP 配置行。配置更改如下所示。

中心路由器（旧）

```
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.<n>.1 set
transform-set trans2 match address <n+100> ! 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<n> bandwidth 1000 ip address
10.0.0.<4n-1> 255.255.255.252 ip mtu 1400 delay 1000
tunnel source Ethernet0 tunnel destination 172.16.<n>.1
! 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 <n+100> permit gre host 172.17.0.1 host
172.16.<n>.1
```

中心路由器（新）

```
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
```

Spoke<n> 路由器（旧）

```
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> 路由器（新）

```
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.<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 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 !
```

在分支路由器上，子网掩码已更改，并且已在隧道接口下添加 NHRP 命令。NHRP 命令是必需的，因为中心路由器目前使用 NHRP 将分支隧道接口 IP 地址映射到分支物理接口 IP 地址。

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 ... tunnel key 100000

此时子网是 /24 而不是 /30，因此所有节点均在同一子网内，而不是位于不同的子网。分支仍然通过中心发送分支到分支的数据流，因为它们使用的是点对点 GRE 隧道接口。**ip nhrp authentication ...**、**ip nhrp network-id ...** 和 **tunnel key ...** 命令用于在中心路由器收到隧道数据包和 NHRP 数据包时将这些数据包映射到正确的多点 GRE 隧道接口和 NHRP 网络。**ip nhrp map ...** 和 **ip nhrp nhs ...** 命令由分支上的 NHRP 用于向中心广播分支 NHRP 映射 (10.0.0.<n+1>--> 172.16.<n>.1)。10.0.0.<n+1> 地址通过隧道接口的 **ip address ...** 命令进行检索，172.16.<n>.1 地址通过隧道接口的 **tunnel destination ...** 命令进行检索。

如果有 300 个分支路由器，此更改会使中心路由器上的配置行数从 3900 行减为 16 行（减少 3884 行）。每个分支路由器上的配置将增加 6 行。

在分支上支持动态地址

在 Cisco 路由器上，必须先为每个 IPsec 隧道配置另一 IPsec 对等体的 IP 地址，才能建立 IPsec 隧道。如果分支路由器的物理接口具有动态地址，此操作将会遇到问题，而这种情况对于通过 DSL 或电缆链路连接的路由器十分常见。

TED 允许一个 IPsec 对等体查找另一个 IPsec 对等体，方法是将专用的 Internet 安全连接和密钥管理协议 (ISAKMP) 数据包发送到需要加密的原始数据包的 IP 目标地址。假设前提是此数据包将沿着 IPsec 隧道数据包所采用的路径通过中间网络。另一端的 IPsec 对等体将接收此数据包，并对前一个对等体做出响应。随后两个路由器将协商 ISAKMP 和 IPsec 安全连接 (SAS) 并建立 IPsec 隧道。要加密的数据包必须具有可路由的 IP 地址，上述操作才有效。

TED 可与 GRE 隧道结合使用，配置如前一部分所示。此配置已经过测试并且可以使用，但 Cisco IOS 软件的早期版本中存在一个错误，即 TED 会对两个 IPsec 对等体之间的所有 IP 数据流（而不仅仅是 GRE 隧道数据包）强制加密。DMVPN 解决方案可以在没有必须使用 Internet 可路由 IP 地址的主机的情况下提供此功能和其他功能，而且无需发送探测器和响应数据包。只需稍加修改，上一部分的配置便可用于支持外部物理接口具有动态 IP 地址的分支路由器。

中心路由器（无更改）

```
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
```

Spoke<n> 路由器（旧）

```
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.<n>.1 host
172.17.0.1
```

Spoke<n> 路由器（新）

```
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
```

新分支配置中使用的功能如下。

- GRE 隧道接口启动时，便会开始将 NHRP 注册数据包发送到中心路由器。这些 NHRP 注册数据包将触发 IPsec 隧道的建立。在分支路由器上配置了 set peer <peer-address> 和 **match ip access-list <ACL>** 命令。ACL 指定 GRE 作为协议，any 作为源，中心路由器 IP 地址作为目标。**注意：** 请注意很重要的一点，ACL 中使用 any 作为源，而且必须如此，因为分支路由器的 IP 地址是动态地址，在物理接口启动之前该地址是未知的。如果动态分支接口地址将限制为 IP 子网内的某个地址，则可以使用该子网作为 ACL 中的源。
- 使用 **set security-association level per-host** 命令，分支 IPsec 代理中的 IP 源地址将是分支当前物理接口地址 (/32)，而不是 ACL 中的“any”。如果使用 ACL 中的“any”作为 IPsec 代理的源，则会阻止任何其他分支路由器建立连接此中心路由器的 IPsec+GRE 隧道。这是因为这样一来，中心路由器上的 IPsec 代理将等同于 **permit gre host 172.17.0.1 any**。这意味着所有发往任何分支的 GRE 隧道数据包都将加密并发送到建立连接中心路由器的隧道的第一个分支，因为其 IPsec 代理与每个分支的 GRE 数据包均匹配。
- 一旦建立 IPsec 隧道，NHRP 注册数据包便会从分支路由器发往配置的下一跳服务器 (NHS)。NHS 是此星型网络的中心路由器。NHRP 注册数据包为中心路由器提供信息，用于创建此分支路由器的 NHRP 映射。使用此映射，中心路由器便可以通过 mGRE+IPsec 隧道将单播 IP 数据

包转发至此分支路由器。另外，中心路由器还会将分支路由器添加到其 NHRP 组播映射列表中。接着中心会开始将动态 IP 路由组播数据包发送到分支（前提是配置了动态路由协议）。随后分支将成为中心的路由协议邻居，它们将交换路由更新。

IPsec + mGRE 星型网络

中心路由器

```
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 !
```

请注意，上述中心路由器配置中没有配置分支路由器的 IP 地址。分支的外部物理接口和到分支隧道接口 IP 地址的映射是由中心路由器通过 NHRP 动态学习的。这样，便可以动态分配分支的外部物理接口 IP 地址。

Spoke1 路由器

```
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
```

Spoke2 路由器

```
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
```

关于分支配置，需要注意以下几点：

- 外部物理接口 (ethernet0) IP 地址通过 DHCP 动态分配。ip address dhcp hostname Spoke2
- 加密 ACL (101) 指定子网作为 IPsec 代理的源。access-list 101 permit gre 172.16.2.0 0.0.0.255 host 172.17.0.1
- IPsec 加密映射中的以下命令指定安全连接将以每个主机为单位。set security-association level per-host
- 所有隧道都是同一子网的一部分，因为它们全部通过中心路由器上的同一个多点 GRE 接口进行连接。ip address 10.0.0.2 255.255.255.0

组合使用这三个命令必须配置分支的外部物理接口 IP 地址。使用的 IPsec 代理将基于主机而不是基于子网。

在分支路由器上配置了中心路由器的 IP 地址，因为它需要建立 IPsec+GRE 隧道。请注意 Spoke1 配置与 Spoke2 配置之间的相似性。不仅这两个配置具有相似性，所有分支路由器配置都将是类似的。在大多数情况下，所有分支只需要具有唯一的接口 IP 地址，它们的其余配置将是相同的。这样，便可以迅速配置和部署许多分支路由器。

中心和分支上的 NHRP 数据看起来与下图类似。

中心路由器

```
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
```

Spoke1 路由器

```
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
```

动态多点星型结构

上述分支路由器配置并不依靠 DMVPN 解决方案的功能，因此这些分支路由器可以运行 12.2(13)T 之前的 Cisco IOS 软件版本。但中心路由器上的配置依靠 DMVPN 功能，因此必须运行 Cisco IOS 版本 12.2(13)T 或更高版本。这样，当您需要升级已部署的分支路由器时，可以在决策方面获得一定的灵活性。如果您的分支路由器同样运行 Cisco IOS 版本 12.2(13)T 或更高版本，则可以如下简化分支配置。

Spoke<n> 路由器（版本低于 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> 路由器（版本高于 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> !
```

请注意，我们已经完成了下列操作：

1. 删除 **crypto map vpnmap1 10 ipsec-isakmp** 命令并将其替换为 **crypto ipsec profile vpnprof**。
2. 从 Ethernet0 接口删除 **crypto map vpnmap1** 命令，并在 Tunnel0 接口应用 **tunnel protection ipsec profile vpnprof** 命令。
3. 删除加密 ACL **access-list 101 permit gre any host 172.17.0.1**。

在这种情况下，IPsec 对等体地址和代理将自动从 **tunnel source ...** 和 **tunnel destination ...** 配置中生成。对等体和代理如下所示（在 **show crypto ipsec sa** 命令的输出中显示）：

```
...
```

```
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
...
```

综上所述，以下完整配置包括了到目前为止以基本配置（IPsec+GRE 星型网络）为起点所做的所有更改。

中心路由器

```
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
 !
```

中心路由器配置没有更改。

Spoke1 路由器

```
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
!
```

Spoke2 路由器

```
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
!
```

动态多点 IPsec VPN

本部分的概念和配置显示了 DMVPN 的完整功能。NHRP 为分支路由器提供动态学习 VPN 网络中其他分支路由器的外部物理接口地址的功能。这意味着分支路由器获得的信息将足以用于动态建立直接连接其他分支路由器的 IPsec+mGRE 隧道。这非常有利，因为如果这种分支到分支的数据流是通过中心路由器发送的，就必须对其进行加密/解密，使中心路由器的延迟和负载增加两次。为使用此功能，分支路由器需要从点对点 GRE (p-pGRE) 切换到多点 GRE (mGRE) 隧道接口。另外，它们还需要学习具有另一分支路由器隧道 IP 地址的 IP 下一跳的其他分支后面的可用（子）网络。分支路由器通过跨连接中心路由器的 IPsec+mGRE 隧道运行的动态 IP 路由协议学习这些（子）网络。

在中心路由器上运行的动态 IP 路由协议可以配置为将从一个分支学习的路由通过同一个接口返回到所有其他分支，但这些路由的 IP 下一跳通常将是中心路由器，而不是中心路由器学习此路由的来源分支路由器。

注意： 动态路由协议仅在中心与分支链路上运行，而不在动态分支到分支链路上运行。

需要在中心路由器上配置动态路由协议（RIP、OSPF 和 EIGRP）以通过 mGRE 隧道接口返回路

由广播，并在路由广播返回其他分支时将 IP 下一跳设置为学习路由的来源分支路由器。

下面是路由协议配置的要求。

RIP

您需要在中心路由器上关闭 mGRE 隧道接口的水平分割，否则 RIP 不会将通过 mGRE 接口学习的路由通过同一接口广播返回。

no ip split-horizon
不需要其他更改。RIP 将自动使用它在学习路由的接口广播返回的路由的原始 IP 下一跳。

EIGRP

您需要在中心路由器上关闭 mGRE 隧道接口的水平分割，否则 EIGRP 不会将通过 mGRE 接口学习的路由通过同一接口广播返回。

no ip split-horizon eigrp <as>
默认情况下，EIGRP 会将 IP 下一跳设置为所广播路由的中心路由器，即使是在学习路由的接口将这些路由广播返回也是如此。因此，在这种情况下，需要使用以下配置命令指示 EIGRP 在广播这些路由时使用原始 IP 下一跳。

no ip next-hop-self eigrp <as>
注意： **no ip next-hop-self eigrp <as>** 命令将从 Cisco IOS 版本 12.3(2) 开始提供。对于 12.2(13)T 和 12.3(2) 之间的 Cisco IOS 版本，则必须执行下列操作：

- 如果不需要分支到分支的动态隧道，则不需要上述命令。
- 如果需要分支到分支的动态隧道，则必须在分支路由器的隧道接口使用进程交换。
- 否则，将必须跨 DMVPN 使用不同的路由协议。

OSPF

由于 OSPF 是链路状态路由协议，因此不存在任何水平分割问题。对于多点接口，通常会将 OSPF 网络类型配置为点对多点，但这会导致 OSPF 在分支路由器的路由表中添加主机路由。这些主机路由将导致发往其他分支路由器后面网络的数据包通过中心路由器进行转发，而不是直接转发至其他分支。要避免此问题，请使用此命令将 OSPF 网络类型配置为广播。

ip ospf network broadcast
此外，还需要确保中心路由器将是 IPsec+mGRE 网络的指定路由器 (DR)。这一点的实现方法是在中心路由器上将 OSPF 优先级设置为大于 1，在分支路由器上将其设置为 0。

- 集线器：**ip ospf priority 2**
- 分支：**ip ospf priority 0**

DMVPN 单中心

中心路由器
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 !
```

中心路由器配置的唯一更改是 OSPF 变为路由协议而不是 EIGRP。请注意，OSPF 网络类型设置为广播，优先级设置为 2。将 OSPF 网络类型设置为广播会导致 OSPF 将具有 IP 下一跳地址的分支路由器后面网络的路由设置为该分支路由器的 GRE 隧道地址。

Spoke1 路由器

```
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 !
```

此时分支路由器上的配置与中心路由器上的配置非常相似。区别如下：

- OSPF 优先级设置为 0。不能允许分支路由器成为 mGRE 非广播多路访问 (NBMA) 网络的 DR。只有中心路由器拥有与所有分支路由器的直接静态连接。DR 必须能够访问 NBMA 网络的所有成员。
- 为中心路由器配置了 NHRP 单播和组播映射。

  ip nhrp map multicast 172.17.0.1 ip nhrp map 10.0.0.1 172.17.0.1 在先前的配置中不需要 **ip nhrp map multicast ...** 命令，因为 GRE 隧道是点对点隧道。在那种情况下，将自动封装组播数据包并通过隧道发送到可能的单个目标。现在需要使用此命令，因为分支 GRE 隧道已更改为多点隧道，可能的目标不止一个。
- 当分支路由器启动时，它必须与中心路由器建立隧道连接，因为中心路由器没有配置任何与分支路由器有关的信息，而且分支路由器可能具有动态分配的 IP 地址。分支路由器也将中心路由器配置为它们的 NHRP NHS。`ip nhrp nhs 10.0.0.1`使用上述命令，分支路由器会以固定间隔通过 mGRE+IPsec 隧道将 NHRP 注册数据包发送到中心路由器。这些注册数据包将提供中心路由器通过隧道将数据包返回分支路由器所需的分支 NHRP 映射信息。

Spoke2 路由器

```
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 !
```

Spoke<n> 路由器

```
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.<n>.0 0.0.0.255 area 0 !
```

请注意，所有分支路由器的配置都很相似。唯一不同的是本地接口的 IP 地址。这在部署大量分支路由器时很有帮助。可以对所有分支路由器进行相同配置，只需添加本地 IP 接口地址即可。

这时请观察中心路由器、Spoke1 和 Spoke2 路由器上的路由表和 NHRP 映射表以查看初始状况（Spoke1 和 Spoke2 刚启动时的状况），以及 Spoke1 和 Spoke2 在它们之间创建动态链路之后的状况。

初始状况

中心路由器信息

```
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
```

Spoke1 路由器信息

```
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
```

Spoke2 路由器信息

```
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
```

此时从 192.168.1.2 向 192.168.2.3 执行 ping 命令。这些地址分别是 Spoke1 和 Spoke2 路由器后面的主机的地址。下列事件依次发生以建立分支到分支的直接 mGRE+IPsec 隧道。

1. Spoke1 路由器收到目标为 192.168.2.3 的 ping 数据包。它在路由表中查找此目标，并发现需要通过 Tunnel0 接口将此数据包转发至 IP 下一跳 10.0.0.3。
2. Spoke1 路由器在 NHRP 映射表中查找目标 10.0.0.3，发现没有此条目。Spoke1 路由器创建 NHRP 解析请求数据包，并将该数据包发送到其 NHS（中心路由器）。
3. 中心路由器在其 NHRP 映射表中查找目标 10.0.0.3，发现它映射到地址 172.16.2.75。中心路由器创建 NHRP 解析应答数据包，并将其发送到 Spoke1 路由器。
4. Spoke1 路由器收到 NHRP 解析应答，并在其 NHRP 映射表中输入 10.0.0.3 — > 172.16.2.75 映射。NHRP 映射的添加触发 IPsec 建立连接对等体 172.16.2.75 的 IPsec 隧道。
5. Spoke1 路由器与 172.16.2.75 建立 ISAKMP 并协商 ISAKMP 和 IPsec SA。通过 Tunnel0 的 tunnel source <address> 命令和 NHRP 映射生成 IPsec 代理。
   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. IPsec 隧道建立完成后，所有发往 192.168.2.0/24 子网的后续数据包都将直接发送到 Spoke2。
7. 在发往 192.168.2.3 的数据包转发至主机后，此主机会向 192.168.1.2 发送一个返回数据包。当 Spoke2 路由器收到发往 192.168.1.2 的这个数据包时，它会在路由表中查找此目标，并发现需要通过 Tunnel0 接口将此数据包转发至 IP 下一跳 10.0.0.2。
8. Spoke2 路由器在 NHRP 映射表中查找目标 10.0.0.2，发现没有此条目。Spoke2 路由器创建 NHRP 解析请求数据包，并将该数据包发送到其 NHS（中心路由器）。
9. 中心路由器在其 NHRP 映射表中查找目标 10.0.0.2，发现它映射到地址 172.16.1.24。中心路由器创建 NHRP 解析应答数据包，并将其发送到 Spoke2 路由器。
10. Spoke2 路由器收到 NHRP 解析应答，并在其 NHRP 映射表中输入 10.0.0.2 — > 172.16.1.24 映射。NHRP 映射的添加触发 IPsec 建立连接对等体 172.16.1.24 的 IPsec 隧道，但是连接对等体 172.16.1.24 的 IPsec 隧道已经存在，因此不需要任何进一步的操作。
11. 此时 Spoke1 和 Spoke2 便可以相互直接转发数据包了。如果在 holdtime 内未使用 NHRP 映

射转发数据包，则会删除该 NHRP 映射。删除 NHRP 映射条目将触发 IPsec 删除此直接链接的 IPsec SA。

在 Spoke1 和 Spoke2 之间创建动态链路之后的状况

Spoke1 路由器信息

```
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
```

Spoke2 路由器信息

```
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
```

通过以上输出可以看到，Spoke1 和 Spoke2 已从中心路由器获得彼此的 NHRP 映射，并已建立和使用 mGRE+IPsec 隧道。NHRP 映射将于五分钟（NHRP holdtime 的当前值为 300 秒）后到期。如果在到期前最后一分钟内使用了 NHRP 映射，则会发送 NHRP 解析请求和应答以便在删除条目之前对其进行刷新。否则，将删除 NHRP 映射，这会触发 IPsec 清除 IPsec SA。

使用双集线器的动态多点 IPsec VPN

只需为分支路由器添加几行配置，便可以设置双（或多个）中心路由器以实现冗余。配置双中心 DMVPN 的方法有两种。

- 单 DMVPN 网络中的每个分支使用单个多点 GRE 隧道接口并指向两个不同的中心作为其下一跳服务器 (NHS)。中心路由器将只有一个多点 GRE 隧道接口。
- 而在双 DMVPN 网络中，每个分支有两个 GRE 隧道接口（点对点或多点），每个 GRE 隧道连接到不同的中心路由器。同样，中心路由器将只有一个多点 GRE 隧道接口。

下面的示例将显示如何配置这两种不同的双中心 DMVPN 方案。在这两种情况下，突出显示的区别是相对于 DMVPN 单中心配置的区别。

双集线器 - 单 DMVPN 布局

双中心单 DMVPN 布局的设置非常简单，但您无法获得与双中心双 DMVPN 布局一样多的路由控制权。这里的意图是使用一个包含所有中心路由器（本例中是两个）的 DMVPN“云”，所有分支路由器

均连接到这个子网（“云”）。从分支到中心的静态 NHRP 映射定义了运行动态路由协议的静态 IPsec+mGRE 链路。动态路由协议不会在分支之间的动态 IPsec+mGRE 链路上运行。由于分支路由器是中心路由器在同一 mGRE 隧道接口的路由邻居，因此不能利用链路或接口差异（如度量标准、成本、延迟或带宽）修改动态路由协议指标以便在两个中心路由器均启用时首选其中一个中心路由器。如果需要此首选项，则必须使用路由协议配置的内部技术。因此，使用 EIGRP 或 RIP（而不是 OSPF）作为动态路由协议可能更好。

注意： 通常只有中心路由器位置相同时，才需要解决上述问题。如果它们位置不同，正常动态路由最终很可能会选择正确的中心路由器，即使通过任一中心路由器均可到达目标网络也是如此。

双集线器 - 单 DMVPN 布局

中心路由器

```
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 !
```

Hub2 路由器

```
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 !
```

Hub1 配置的唯一更改是将 OSPF 更改为使用两个区域。区域 0 用于两个中心路由器后面的网络，区域 1 用于 DMVPN 网络和分支路由器后面的网络。OSPF 可以使用单个区域，但此处使用两个区域以展示多个 OSPF 区域的配置。

Hub2 的配置与 Hub1 的配置基本相同，只需进行适当的 IP 地址更改即可。主要区别在于 Hub2 也是 Hub1 的分支（或客户端），因此 Hub1 是主中心，Hub2 是辅助中心。此设置是通过 mGRE 隧道使 Hub2 成为 Hub1 的 OSPF 邻居。由于 Hub1 是 OSPF DR，因此必须通过 mGRE 接口直接连接到所有其他 OSPF 路由器（NBMA 网络）。如果 Hub1 和 Hub2 之间没有直接链接，则当 Hub1 也启用时，Hub2 不会参与 OSPF 路由。当 Hub1 发生故障时，Hub2 将成为 DMVPN 的 OSPF DR（NBMA 网络）。Hub1 恢复后，它将接管 DMVPN 的 OSPF DR 角色。

Hub1 和 Hub2 后面的路由器将使用 Hub1 向分支网络发送数据包，因为 GRE 隧道接口的带宽设置为 1000 Kb/秒，而 Hub2 上的带宽为 900 Kb/秒。相反，分支路由器会将中心路由器后面网络的数据包同时发送到 Hub1 和 Hub2，因为每个分支路由器上只有一个 mGRE 隧道接口，却有两个成本相等的路由。如果使用基于数据包的负载均衡，这会产生无序数据包。

Spoke1 路由器

```
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 !
```

分支路由器配置的区别如下：

- 在新的配置中，分支配置了 Hub2 的静态 NHRP 映射，并将 Hub2 添加为下一跳服务器。原始

：

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 新：**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**

- 分支路由器上的 OSPF 区域已更改为区域 1。

请记住，通过在分支路由器上定义中心路由器的静态 NHRP 映射和 NHS，动态路由协议将在此隧道上运行。此操作定义了中心和分支路由或邻居网络。请注意，Hub2 是所有分支的中心，也是 Hub1 的分支。这样，当您使用 DMVPN 解决方案时，便可以轻松设计、配置和修改多层星型网络。

Spoke2 路由器

```
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 !
```

Spoke<n> 路由器

```
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.<n>.0 0.0.0.255 area 0 !
```

此时，可以观察 Hub1、Hub2、Spoke1 和 Spoke2 路由器上的路由表、NHRP 映射表和 IPsec 连接，以查看初始状况（Spoke1 和 Spoke2 路由器刚刚启动时）。

初始状况和更改

Hub1 路由器信息

```
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
```

Hub2 路由器信息

```
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
```

Spoke1 路由器信息

```
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
```

Spoke2 路由器信息

```
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
```

下面是有关 Hub1、Hub2、Spoke1 和 Spoke2 上路由表的几个有趣的问题：

- 两个中心路由器具有通往分支路由器后面网络的成本相等的路由。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
```
这意味着 Hub1 和 Hub2 会针对分支路由器后面的网络向中心路由器后面网络中的路由器广播相同成本。例如，直接连接到 192.168.0.0/24 LAN 的路由器上的路由表 R2 将类似下面所示：
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, Ethernet1/0/3
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
```

- 分支路由器具有通过两个中心路由器通往其后面网络的成本相等的路由。
分支1 ：
```
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
```
分支2 ：
```
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
```
如果分支路由器当前执行基于数据包的负载均衡，则可能会得到无序数据包。

要避免通过两个中心路由器之间的链路进行非对称路由或基于数据包的负载均衡，需要将路由协议配置为在两个方向均选择同一个分支到中心的路径。如果希望 Hub1 作为主中心，Hub2 作为备份中心，则可以将中心隧道接口的 OSPF 成本设置为不同值。

Hub1 ：

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

Hub2 ：

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

现在路由如下所示：

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
```

此时，两个中心路由器对于分支路由器后面网络的路由具有不同成本。这意味着将首选 Hub1 用于将数据流转发至分支路由器，在路由器 R2 上可以看出这一点。这样便可以解决上面第一项中说明的非对称路由问题。

另一个方向的非对称路由（如上面第二项所述）则仍然存在。使用 OSPF 作为动态路由协议时，可以采用一种应急方案解决此问题，即在分支上使用 **router ospf 1** 下的 **distance ...** 命令以首选通过 Hub1 学习的路由（通过 Hub2 学习的路由为备选）。

分支1 ：

```
router ospf 1
 distance 111 10.0.0.2 0.0.0.0 1
access-list 1 permit any
```
分支2 ：

```
router ospf 1
 distance 111 10.0.0.2 0.0.0.0 1
access-list 1 permit any
```
现在路由如下所示：

分支1 ：

```
O     192.168.0.0/24 [110/11] via 10.0.0.1, 00:00:06, Tunnel0
```
分支2 ：

```
O     192.168.1.0/24 [110/11] via 10.0.0.1, 00:00:10, Tunnel0
```
上述路由配置将防止受到非对称路由的危害，同时允许在 Hub1 发生故障时故障切换至 Hub2。这意味着，当两个中心路由器均启用时，仅使用 Hub1。如果要通过平衡跨中心分支同时使用两个中心路由器，并提供故障切换保护和消除非对称路由，路由配置可能会比较复杂，特别是使用 OSPF 时。为此，下面的双中心双 DMVPN 布局可能是更好的选择。

双中心 - 双 DMVPN 布局

双中心双 DMVPN 布局的设置稍难一点，但可以为您提供更好的跨 DMVPN 路由控制。意图是使用两个单独的 DMVPN“云”。每个中心路由器（本例中是两个）连接到一个 DMVPN 子网（“云”），分支路由器连接到两个 DMVPN 子网（“云”）。由于分支路由器是两个中心路由器通过两个 GRE 隧道的路由邻居，因此可以利用接口差异（如带宽、成本和延迟）修改动态路由协议指标以便在两个中心路由器均启用时首选其中一个中心路由器。

注意： 通常只有中心路由器位置相同时，才需要解决上述问题。如果它们位置不同，正常动态路由最终很可能会选择正确的中心路由器，即使通过任一中心路由器均可到达目标网络也是如此。

可以在分支路由器上使用 p-pGRE 或 mGRE 隧道接口。一个分支路由器上的多个 p-pGRE 接口可以使用相同的 **tunnel source ...** IP 地址，但一个分支路由器上的多个 mGRE 接口必须具有唯一的 **tunnel source ...** IP 地址。这是因为当 IPsec 隧道建立时，第一个数据包是需要与其中一个 mGRE 隧道关联的 ISAKMP 数据包。该 ISAKMP 数据包只有可用于建立关联的目标 IP 地址（远程 IPsec 对等体地址）。此地址将与 **tunnel source ...** 地址进行匹配，但由于两个隧道具有相同的 **tunnel source ...** 地址，因此第一个 mGRE 隧道接口始终与之匹配。这意味着传入组播数据包可能会与错误的 mGRE 接口建立关联，从而导致所有动态路由协议中断。

GRE 数据包本身不存在此问题，因为它们的 **tunnel key ...** 值可以区分两个 mGRE 接口。从 Cisco IOS 软件版本 12.3(5) 和 12.3(7)T 开始，引入了一个附加参数以消除此限制：**tunnel protection....共享。shared** 关键字表示多个 mGRE 接口将使用具有相同源 IP 地址的 IPsec 加密。如果使用早期版本，您可以在这种双中心双 DMVPN 布局中使用 p-pGRE 隧道。在使用 p-pGRE 隧道的情况下，**tunnel source ...** 和 **tunnel destination ...** IP 地址可用于进行匹配。对于本例，将在此双中心双 DMVPN 布局中使用 p-pGRE 隧道，且不使用 **shared** 限定符。

双中心 - 双 DMVPN 布局

下面突出显示的更改是相对于本文档前面介绍的动态多点星型网络配置的更改。

Hub1 路由器
`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 !
```

Hub2 路由器

```
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 !
```

在这种情况下，Hub1 和 Hub2 的配置十分相似。主要区别在于两者分别是不同 DMVPN 的中心路由器。每个 DMVPN 使用不同的：

- IP 子网（10.0.0.0/24、10.0.0.1/24）
- NHRP 网络 ID（100000、100001）
- 隧道密钥（100000、100001）

动态路由协议已从 OSPF 切换为 EIGRP，因为使用 EIGRP 建立和管理 NBMA 网络更加容易，本文档稍后将进行介绍。

Spoke1 路由器

```
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 !
```

每个分支路由器配置两个 p-pGRE 隧道接口，每个 DMVPN 一个。**ip address ...**、**ip nhrp network-id ...**、**tunnel key ...** 和 **tunnel destination ...** 值用于区分两个隧道。动态路由协议 EIGRP 同时在两个 p-pGRE 隧道子网上运行，用于在两个 p-pGRE 接口 (DMVPN) 中首选一个接口。

Spoke2 路由器

```
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 !
```

Spoke<n> 路由器

```
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 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.<n+10> 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 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 !
```

此时观察 Hub1、Hub2、Spoke1 和 Spoke2 路由器上的路由表、NHRP 映射表和 IPsec 连接，以查看初始状况（Spoke1 和 Spoke2 路由器刚刚启动时）。

初始状况和更改

Hub1 路由器信息

```
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
```

Hub2 路由器信息

```
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
```

Spoke1 路由器信息

```
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
```

Spoke2 路由器信息

```
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
```

同样，下面是有关 Hub1、Hub2、Spoke1 和 Spoke2 上路由表的几个有趣的问题：

- 两个中心路由器具有通往分支路由器后面网络的成本相等的路由。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`这意味着 Hub1 和 Hub2 会针对分支路由器后面的网络向中心路由器后面网络中的路由器广播相同成本。例如，直接连接到 192.168.0.0/24 LAN 的路由器上的路由表 R2 将类似下面所示：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`
- 分支路由器具有通过两个中心路由器通往其后面网络的成本相等的路由。分支1 ：`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`分支2 ：`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`如果分支路由器当前执行基于数据包的负载均衡，则可能会得到无序数据包。

要避免通过两个中心路由器之间的链路进行非对称路由或基于数据包的负载均衡，需要将路由协议配置为在两个方向均选择同一个分支到中心的路径。如果希望 Hub1 作为主中心，Hub2 作为备份中心，则可以将中心隧道接口的延迟设置为不同值。

Hub1 ：

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

Hub2 ：

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

注意： 在本例中，将 Hub2 隧道接口的延迟增加了 50，因为它小于两个中心路由器之间的 Ethernet1 接口的延迟 (100)。这样，Hub2 仍会将数据包直接转发至分支路由器，但它会向 Hub1 和 Hub2 后面的路由器广播首选性低于 Hub1 的路由。如果延迟增加量超过 100，Hub2 将使用 Ethernet1 接口通过 Hub1 转发分支路由器的数据包，但 Hub1 和 Hub2 后面的路由器仍将正确选择使用 Hub1 将数据包发送到分支路由器。

现在路由如下所示：

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
```

两个中心路由器在分支路由器后面的网络路由成本不同，因此，在这种情况下，将首选使用 Hub1 将数据流转发至分支路由器，如 R2 所示。这样可以解决上面第一项所述的问题。

上面第二项所述的问题仍然存在，但由于有两个 p-pGRE 隧道接口，因此可以分别设置隧道接口上的 **delay ...** 以更改通过 Hub1 与 Hub2 学习的路由的 EIGRP 指标。

分支1 ：

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

分支2 ：

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

现在路由如下所示：

分支1 ：

```
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
```

分支2 ：

```
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
```

上述路由配置将防止受到非对称路由的危害，同时允许在 Hub1 发生故障时故障切换至 Hub2。这意味着，当两个中心路由器均启用时，仅使用 Hub1。

如果要通过平衡跨中心分支同时使用两个中心路由器，并提供故障切换保护和消除非对称路由，路由配置可能会比较复杂，但使用 EIGRP 时可以实现这一点。为此，请将中心路由器上隧道接口的 **delay ...** 重新设置为相等值，然后对分支路由器使用 **offset-list <acl> out <offset> <interface>** 命令，针对通过 GRE 隧道接口向备份中心路由器广播的路由增大 EIGRP 指标。分支上的 Tunnel0 和 Tunnel1 接口之间仍然使用不相等的 **delay ...** 值，因此分支路由器将首选其主中心路由器。分支路由器的更改如下。

Spoke1 路由器

```
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 !
```

Spoke2 路由器

```
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 !
```

注意： 为 EIGRP 指标增加了偏移值 12800 (50*256)，因为它小于 25600 (100*256)。此值 (25600) 是针对中心路由器之间学习的路由为 EIGRP 指标添加的值。通过在 **offset-list** 命令中使用 12800，备用中心路由器会将数据包直接转发至分支路由器，而不是使用以太网通过主中心路由器为分支转发这些数据包。中心路由器广播的路由的指标仍将是用于首选正确的主中心路由器的值。请记住，有一半分支将 Hub1 作为其主路由器，另外一半将 Hub2 作为其主路由器。

注意： 如果增加的偏移值超过 25600 (100*256)，中心路由器会使用 Ethernet1 接口通过另一个中心路由器转发一半分支路由器的数据包，但中心路由器后面的路由器仍将首选使用正确的中心路由器将数据包发送到分支路由器。

注意： 此外，还添加了 **distribute-list 1 out** 命令，因为通过分支上某个隧道接口从一个中心路由器学习的路由可能会通过另一隧道重新广播到另一个中心路由器。**distribute-list ...** 命令可确保分支路由器只能广播自己的路由。

注意： 如果希望控制中心路由器（而不是分支路由器）的路由广播，则可以在中心路由器（而不是分支路由器）上配置 **offset-list <acl1> in <value> <interface>** 和 **distribute-list <acl2> in** 命令。<acl2> 访问列表将列出从所有分支后面出发的路由，<acl1> 访问列表将仅列出从将另一中心路由器

作为主中心的分支后面出发的路由。

经过上述更改，路由将如下所示：

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
```

分支1 ：

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

分支2 ：

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

结论

DMVPN 解决方案可提供以下功能，用于改进大型和小型 IPsec VPN 网络的规模调整。

- DMVPN 可在完全网状或部分网状 IPsec VPN 中实现更好的规模调整。当分支到分支的数据流为间歇性（例如，每个分支并非不断向所有其他分支发送数据）时，此解决方案尤其适用。只要分支之间存在直接 IP 连接，它就允许任何分支向任何其他分支直接发送数据。
- DMVPN 支持具有动态分配地址（如电缆、ISDN 和 DSL）的 IPsec 节点。这适用于星型网络以及网状网络。DMVPN 可要求中心到分支的链路保持打开状态。
- DMVPN 简化了 VPN 节点的添加。在添加新的分支路由器时，您只需配置该分支路由器并将其插入网络即可（但可能需要在中心路由器上为新分支添加 ISAKMP 授权信息）。中心路由器将以动态方式学习该新分支的信息，动态路由协议会将路由传播到中心路由器和所有其他分支路由器。
- DMVPN 减少了 VPN 中所有路由器所需配置的大小。这同样适用于 GRE+IPsec 仅星型 VPN 网络。
- DMVPN 使用 GRE，因此支持跨 VPN 的 IP 多播和动态路由数据流。这意味着可以使用动态路由协议，而且协议可支持冗余“中心路由器”。此外，还支持组播应用。
- DMVPN 支持分支路由器的 Split Tunneling。

相关信息

- 动态多点 VPN (DMVPN)
- IPSec 支持页面
- 技术支持 - Cisco Systems