Dynamische Multipoint-IPsec-VPNs (mit Multipoint-GRE/NHRP zur Skalierung von IPsec-VPNs)

Inhalt

Einführung

In diesem Dokument werden die Dynamic Multipoint IPsec VPNs (DMVPN) erläutert und es wird erläutert, warum ein Unternehmen sein Netzwerk entwerfen oder migrieren möchte, um diese neue IPsec VPN-Lösung in der Cisco IOS® Software zu nutzen.

Hintergrundinformationen

Unternehmen müssen möglicherweise eine Vielzahl von Websites mit einer Hauptniederlassung und möglicherweise auch untereinander über das Internet verbinden, während sie den Datenverkehr verschlüsseln, um ihn zu schützen. Beispielsweise müssen Einzelhandelsgeschäfte, die zur Inventarisierung und Bestellung eine Verbindung zum Hauptsitz des Unternehmens herstellen müssen, möglicherweise auch mit anderen Niederlassungen im Unternehmen verbunden werden, um die Verfügbarkeit des Produkts zu überprüfen. In der Vergangenheit war die einzige Möglichkeit, die Verbindung herzustellen, die Verwendung eines Layer-2-Netzwerks wie ISDN oder Frame Relay, um alle Komponenten miteinander zu verbinden. Das Einrichten und Bezahlen dieser kabelgebundenen Verbindungen für internen IP-Datenverkehr kann zeit- und kostenintensiv sein. Wenn alle Standorte (einschließlich des Hauptstandorts) bereits über einen relativ günstigen Internetzugang verfügen, kann dieser Internetzugang auch für die interne IP-Kommunikation zwischen den Geschäften und dem Hauptsitz verwendet werden, indem IPsec-Tunnel verwendet werden, um Datenschutz und Datenintegrität zu gewährleisten.

Damit Unternehmen große IPsec-Netzwerke aufbauen können, die ihre Standorte über das Internet verbinden, müssen Sie in der Lage sein, das IPsec-Netzwerk zu skalieren. IPsec verschlüsselt den Datenverkehr zwischen zwei Endpunkten (Peers), und die Verschlüsselung wird von den beiden Endpunkten mithilfe eines gemeinsam genutzten "geheimen" Endpunkts durchgeführt. Da dieses Geheimnis nur von diesen beiden Endpunkten gemeinsam genutzt wird, sind verschlüsselte Netzwerke von Natur aus eine Sammlung von Point-to-Point-Links. Aus diesem Grund ist IPsec ein Punkt-zu-Punkt-Tunnelnetzwerk. Die einfachste Methode zur Skalierung eines großen Point-to-Point-Netzwerks besteht in der Organisation eines Hub-and-Spoke- oder Full-Mesh-Netzwerks. In den meisten Netzwerken befindet sich der Großteil des IP-Datenverkehrs zwischen den Spokes und dem Hub, und nur sehr wenig zwischen den Spokes. Daher ist das Hub-and-Spoke-Design oft die beste Wahl. Dieses Design passt auch zu älteren Frame Relay-Netzwerken, da die Bezahlung von Verbindungen zwischen allen Standorten in diesen Netzwerken sehr teuer war.

Wenn das Internet als Verbindung zwischen Hub und Spoke verwendet wird, haben die Stationen auch direkten Zugang ohne zusätzliche Kosten, aber es war sehr schwierig, wenn nicht unmöglich, ein vollständiges (teilweises) Mesh-Netzwerk einzurichten und/oder zu verwalten. Volle oder teilweise vermaschte Netzwerke sind häufig wünschenswert, da Kosteneinsparungen möglich sind, wenn Spoke-to-Spoke-Datenverkehr direkt durchlaufen und nicht über den Hub übertragen werden kann. Spoke-to-Spoke-Datenverkehr, der den Hub durchquert, verwendet Hub-Ressourcen und kann insbesondere bei der IPsec-Verschlüsselung zu zusätzlichen Verzögerungen führen, da der Hub die eingehenden Pakete von den sendenden Spokes entschlüsseln und anschließend den Datenverkehr erneut verschlüsseln muss, um ihn an den empfangenden Spoke zu senden. Ein weiteres Beispiel, in dem ein direkter Spoke-to-Spoke-Datenverkehr nützlich wäre, ist der Fall, in dem sich zwei Spokes in derselben Stadt befinden und sich der Hub landesweit befindet.

Als IPsec-Hub-and-Spoke-Netzwerke bereitgestellt und größer wurden, wurde es wünschenswerter, dass sie IP-Pakete so dynamisch wie möglich weiterleiten. In den älteren Hub-and-Spoke-Netzwerken von Frame Relay wurde dies durch die Ausführung eines dynamischen Routing-Protokolls wie OSPF oder EIGRP über die Frame-Relay-Verbindungen erreicht. Dies war hilfreich, um die Erreichbarkeit von Spoke-Netzwerken dynamisch zu kommunizieren und Redundanz im IP-Routing-Netzwerk zu unterstützen. Wenn im Netzwerk ein Hub-Router verloren geht, kann ein Backup-Hub-Router automatisch die Netzwerkverbindung zu den Spoke-Netzwerken übernehmen.

Es besteht ein grundlegendes Problem mit IPsec-Tunneln und dynamischen Routing-Protokollen. Dynamische Routing-Protokolle basieren auf IP-Multicast- oder Broadcast-Paketen, IPsec unterstützt jedoch nicht die Verschlüsselung von Multicast- oder Broadcast-Paketen. Die aktuelle Methode zur Lösung dieses Problems ist die Verwendung von GRE-Tunneln (Generic Routing Encapsulation) in Kombination mit der IPsec-Verschlüsselung.

GRE-Tunnel unterstützen die Übertragung von IP-Multicast- und Broadcast-Paketen an das andere Ende des GRE-Tunnels. Das GRE-Tunnelpaket ist ein IP-Unicast-Paket, sodass das GRE-Paket mit IPsec verschlüsselt werden kann. In diesem Szenario funktioniert GRE das Tunneling, und IPsec unterstützt das VPN-Netzwerk als Verschlüsselungskomponente. Wenn GRE-Tunnel konfiguriert werden, müssen die IP-Adressen für die Endpunkte des Tunnels (**Tunnelquelle ...**, **Tunnelziel ...**) vom anderen Endpunkt bekannt sein und über das Internet routbar sein. Das bedeutet, dass der Hub und alle Spoke-Router in diesem Netzwerk über statische, nicht private IP-Adressen verfügen müssen.

Bei Verbindungen über kleine Standorte mit dem Internet ändert sich in der Regel die externe IP-Adresse eines Spokes, sobald er eine Verbindung zum Internet herstellt, da sein Internet Service Provider (ISP) bei jedem Online-Spoke (ADSL) und bei Kabeldiensten dynamisch die externe Schnittstellenadresse (DHCP) bereitstellt. Diese dynamische Zuweisung der "externen Adresse" des Routers ermöglicht es dem ISP, die Nutzung seines Internetadressbereichs zu überzeichnen, da nicht alle Benutzer gleichzeitig online sind. Die Zuweisung einer statischen Adresse für den Spoke-Router an den Anbieter kann erheblich teurer sein. Für die Ausführung eines dynamischen Routing-Protokolls über ein IPsec-VPN ist die Verwendung von GRE-Tunneln erforderlich. Sie verlieren jedoch die Möglichkeit, mit dynamisch zugewiesenen IP-Adressen auf ihren externen physischen Schnittstellen zu kommunizieren.

Die oben genannten Einschränkungen und einige andere werden in den folgenden vier Punkten zusammengefasst:

- IPsec verwendet eine Zugriffskontrollliste (ACL), um festzulegen, welche Daten verschlüsselt werden sollen. Bei jedem Hinzufügen eines neuen (Unter-)Netzwerks hinter einem Spoke- oder Hub muss der Kunde die ACL auf den Hub- und Spoke-Routern ändern. Wenn der SP den Router verwaltet, muss der Kunde den SP benachrichtigen, damit die IPsec-ACL geändert wird, sodass der neue Datenverkehr verschlüsselt wird.
- Bei großen Hub-and-Spoke-Netzwerken kann die Konfiguration auf dem Hub-Router sehr groß werden, soweit sie nicht mehr verwendet werden kann. Ein Hub-Router benötigt beispielsweise bis zu 3.900 Konfigurationslinien, um 300 Spoke-Router zu unterstützen. Dies ist groß genug, dass es schwierig wäre, die Konfiguration anzuzeigen und den Konfigurationsabschnitt zu finden, der für ein aktuelles Problem relevant ist, das gerade gedebuggt wird. Diese Größenkonfiguration kann auch zu groß sein, um in den NVRAM integriert zu werden, und müsste im Flash-Speicher gespeichert werden.
- GRE + IPsec muss die Endpunkt-Peer-Adresse kennen. Die IP-Adressen der Stationen sind über einen eigenen ISP direkt mit dem Internet verbunden und werden häufig so eingerichtet, dass ihre externen Schnittstellenadressen nicht behoben werden. Die IP-Adressen können sich ändern, sobald die Website online ist (über DHCP).
- Wenn die Stationen direkt über das IPsec-VPN miteinander kommunizieren müssen, muss das Hub-and-Spoke-Netzwerk zu einem Full-Mesh werden. Da noch nicht bekannt ist, welche Stationen direkt miteinander kommunizieren müssen, ist ein Full Mesh erforderlich, auch wenn nicht jeder Spoke direkt mit jedem anderen Spoke sprechen muss. Außerdem ist es nicht möglich, IPsec auf einem kleinen Spoke-Router zu konfigurieren, sodass er über eine

direkte Verbindung mit allen anderen Spoke-Routern im Netzwerk verfügt. Daher müssen Spoke-Router möglicherweise leistungsstärkere Router sein.

Die DMVPN-Lösung

Die DMVPN-Lösung verwendet Multipoint GRE (mGRE) und Next Hop Resolution Protocol (NHRP) mit IPsec und einigen neuen Erweiterungen, um die oben genannten Probleme skalierbar zu lösen.

Automatische IPsec-Verschlüsselung Initiation

Wenn die DMVPN-Lösung nicht verwendet wird, wird der IPsec-Verschlüsselungstunnel erst initiiert, wenn Datenverkehr vorhanden ist, der die Verwendung dieses IPsec-Tunnels erfordert. Die Initiierung des IPsec-Tunnels kann 1 bis 10 Sekunden dauern, und der Datenverkehr wird während dieser Zeit unterbrochen. Wenn GRE mit IPsec verwendet wird, enthält die GRE-Tunnelkonfiguration bereits die GRE-Tunnel-Peer-Adresse (**Tunnelziel ...**), die auch die IPsec-Peer-Adresse ist. Beide Adressen sind vorkonfiguriert.

Wenn Sie Tunnel Endpoint Discovery (TED) und dynamische Kryptozuordnungen auf dem Hub-Router verwenden, können Sie vermeiden, dass Sie die IPsec-Peer-Adressen auf dem Hub vorkonfigurieren müssen, aber eine TED-Anfrage und -Antwort müssen gesendet und empfangen werden, bevor die ISAKMP-Aushandlung beginnen kann. Dies sollte nicht erforderlich sein, da bei der Verwendung von GRE die Peer-Quell- und Zieladressen bereits bekannt sind. Sie befinden sich entweder in der Konfiguration oder werden mit NHRP (für Mehrpunkt-GRE-Tunnel) aufgelöst.

Mit der DMVPN-Lösung wird IPsec sofort für Point-to-Point- und Multipoint-GRE-Tunnel ausgelöst. Außerdem müssen keine Krypto-ACLs konfiguriert werden, da diese automatisch von den Quell- und Zieladressen des GRE-Tunnels abgeleitet werden. Die folgenden Befehle werden zum Definieren der IPsec-Verschlüsselungsparameter verwendet. Beachten Sie, dass es keinen **festgelegten Peer** gibt **...** oder **Match-Adresse ...** erforderliche Befehle, da diese Informationen direkt von den zugeordneten GRE-Tunnel- oder NHRP-Zuordnungen abgeleitet werden.

```
crypto ipsec profile
```

```
 set transform-set
```

Der folgende Befehl ordnet dem IPsec-Profil eine Tunnelschnittstelle zu.

```
interface tunnel
```

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

Dynamische Tunnelerstellung für "Spoke-to-Hub"-Verbindungen

Auf dem Hub-Router im DMVPN-Netzwerk werden keine GRE- oder IPsec-Informationen zu einem Spoke konfiguriert. Der GRE-Tunnel des Spoke-Routers wird (über NHRP-Befehle) mit Informationen zum Hub-Router konfiguriert. Wenn der Spoke-Router gestartet wird, initiiert er automatisch den IPsec-Tunnel mit dem Hub-Router, wie oben beschrieben. Anschließend wird NHRP verwendet, um den Hub-Router über die aktuelle IP-Adresse der physischen Schnittstelle zu informieren. Dies ist aus drei Gründen nützlich:

- Wenn dem Spoke-Router die IP-Adresse der physischen Schnittstelle dynamisch zugewiesen ist (z. B. mit ADSL oder CableModem), kann der Hub-Router nicht mit diesen Informationen konfiguriert werden, da er bei jedem erneuten Laden des Spoke-Routers eine neue IP-Adresse der physischen Schnittstelle erhält.
- Die Konfiguration des Hub-Routers wird verkürzt und vereinfacht, da keine GRE- oder IPsec-Informationen über die Peer-Router benötigt werden. All diese Informationen werden dynamisch über NHRP erfasst.
- Wenn Sie dem DMVPN-Netzwerk einen neuen Spoke-Router hinzufügen, müssen Sie die Konfiguration auf dem Hub oder auf einem der aktuellen Spoke-Router nicht ändern. Der neue Spoke-Router wird mit den Hub-Informationen konfiguriert, und beim Start wird er dynamisch beim Hub-Router registriert. Das dynamische Routing-Protokoll leitet die Routing-Informationen für diesen Spoke an den Hub weiter. Der Hub leitet diese neuen Routing-Informationen an die anderen Stationen weiter. Außerdem werden die Routing-Informationen von den anderen Stationen an diesen Spoke weitergeleitet.

Dynamische Tunnelerstellung für Spoke-to-Spoke-Datenverkehr

Wie bereits erwähnt, müssen alle Punkt-zu-Punkt-IPsec-Tunnel (oder IPsec+GRE), die sich derzeit in einem Mesh-Netzwerk befinden, auf allen Routern konfiguriert werden, auch wenn einige/die meisten dieser Tunnel nicht ausgeführt werden oder zu jedem Zeitpunkt benötigt werden. Bei der DMVPN-Lösung ist ein Router der Hub, und alle anderen Router (Spokes) werden mit Tunneln zum Hub konfiguriert. Die Spoke-to-Hub-Tunnel sind ständig verfügbar, und Spokes benötigen keine Konfiguration für direkte Tunnel zu anderen Stationen. Wenn ein Spoke ein Paket an ein anderes Spoke übertragen möchte (z. B. das Subnetz hinter einem anderen Spoke), verwendet er stattdessen NHRP, um die erforderliche Zieladresse des Ziel-Spokes dynamisch zu bestimmen. Der Hub-Router fungiert als NHRP-Server und verarbeitet diese Anforderung für das Quell-Spoke. Die beiden Stationen erstellen dann dynamisch einen IPsec-Tunnel zwischen ihnen (über die einzige mGRE-Schnittstelle) und Daten können direkt übertragen werden. Dieser dynamische Spoke-to-Spoke-Tunnel wird nach einer (konfigurierbaren) Inaktivität automatisch beendet.

Unterstützung von dynamischen Routing-Protokollen

Die DMVPN-Lösung basiert auf GRE-Tunneln, die Tunneling-Multicast-/Broadcast-IP-Pakete unterstützen. Daher unterstützt die DMVPN-Lösung auch dynamische Routing-Protokolle, die über IPsec+mGRE-Tunnel ausgeführt werden. Bisher musste das NHRP explizit die Broadcast-/Multicast-Zuordnung für die Tunnel-Ziel-IP-Adressen konfigurieren, um das GRE-Tunneling von Multicast- und Broadcast-IP-Paketen zu unterstützen. Am Hub benötigen Sie beispielsweise die **IP nhrp map multicast <Spoke-n-addr>** Konfigurationszeile für jedes Spoke. Bei der DMVPN-Lösung sind die Spoke-Adressen nicht im Voraus bekannt, daher ist diese Konfiguration nicht möglich. Stattdessen kann das NHRP so konfiguriert werden, dass jedes Spoke automatisch der Multicast-Zielliste auf dem Hub mit dem Befehl **ip nhrp map multicast dynamic** hinzugefügt wird. Wenn die Spoke-Router ihre Unicast-NHRP-Zuordnung mit dem NHRP-Server (Hub) registrieren, erstellt der NHRP mit diesem Befehl auch eine Broadcast-/Multicast-Zuordnung für diese Spoke. Dadurch müssen die Spoke-Adressen nicht im Voraus bekannt sein.

Cisco Express Forwarding Fast Switching für mGRE

Derzeit wird der Datenverkehr in einer mGRE-Schnittstelle prozessgesteuert, was zu einer Leistungsminderung führt. Die DMVPN-Lösung fügt Cisco Express Forwarding Switching für den mGRE-Datenverkehr hinzu, wodurch die Leistung deutlich verbessert wird. Zur Aktivierung dieser Funktion sind keine Konfigurationsbefehle erforderlich. Wenn Cisco Express Forwarding Switching für die GRE-Tunnelschnittstelle und die ausgehenden/eingehenden physischen Schnittstellen zugelassen ist, werden die GRE-Tunnelpakete für mehrere Punkte als Cisco Express Forwarding-Switched (Cisco Express Forwarding-Switched) bezeichnet.

Dynamisches Routing über IPsec geschützte VPNs

In diesem Abschnitt wird der aktuelle Stand der Dinge (vor der DMVPN-Lösung) beschrieben. IPsec wird auf Cisco Routern mithilfe einer Reihe von Befehlen implementiert, die die Verschlüsselung definieren, und anschließend wird ein **Crypto Map <map-name>**-Befehl auf der externen Schnittstelle des Routers angewendet. Aufgrund dieses Designs und der Tatsache, dass es derzeit keinen Standard für die Verwendung von IPsec zur Verschlüsselung von IP-Multicast-/Broadcast-Paketen gibt, können IP-Routing-Protokollpakete nicht über den IPsec-Tunnel "weitergeleitet" werden, und alle Routingänderungen können nicht dynamisch auf die andere Seite des IPsec-Tunnels übertragen werden.

Hinweis: Alle dynamischen Routing-Protokolle außer BGP verwenden Broadcast- oder Multicast-IP-Pakete. GRE-Tunnel werden in Kombination mit IPsec verwendet, um dieses Problem zu lösen.

GRE-Tunnel werden auf Cisco Routern mithilfe einer virtuellen Tunnelschnittstelle implementiert (**Schnittstellentunnel<#>**). Das GRE-Tunneling-Protokoll ist für die Verarbeitung von IP-Multicast-/Broadcast-Paketen konzipiert, sodass ein dynamisches Routing-Protokoll über einen GRE-Tunnel "ausgeführt" werden kann. GRE-Tunnelpakete sind IP-Unicast-Pakete, die das ursprüngliche IP-Multicast-/Unicast-Paket kapseln. Anschließend können Sie IPsec zur Verschlüsselung des GRE-Tunnelpakets verwenden. Sie können auch IPsec im Transportmodus ausführen und 20 Byte speichern, da GRE das ursprüngliche Datenpaket bereits gekapselt hat. Sie benötigen also IPsec nicht, um das GRE IP-Paket in einen anderen IP-Header zu kapseln.

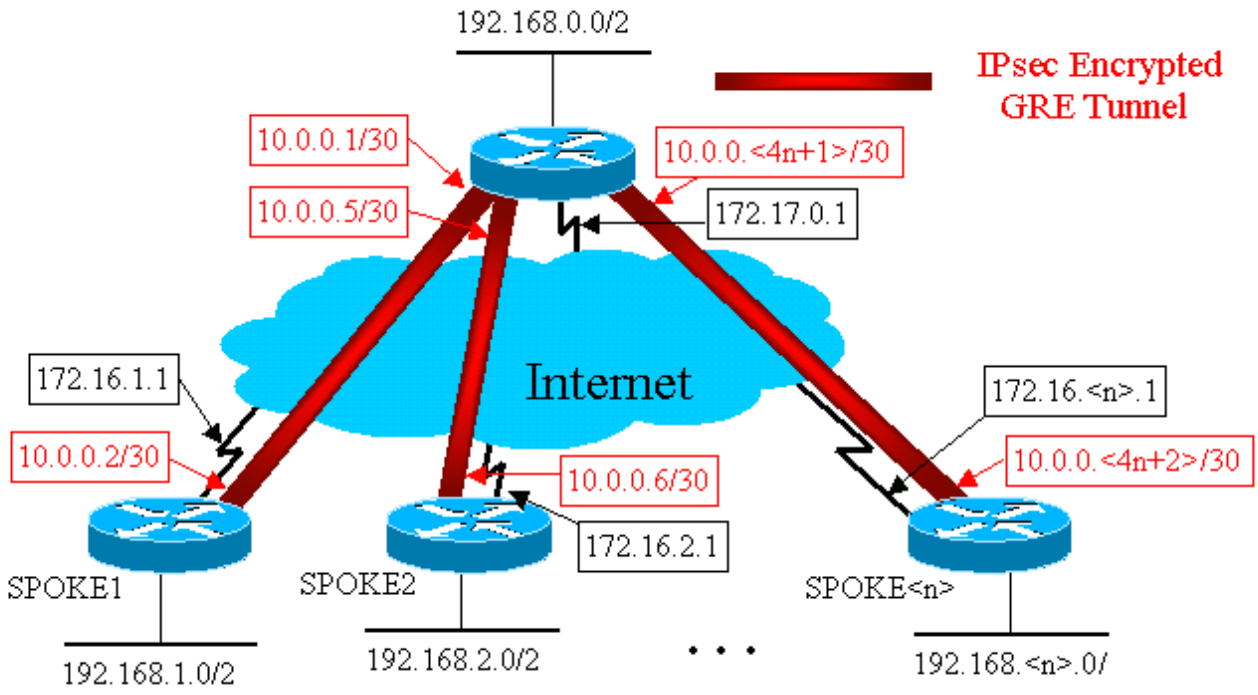
Wenn IPsec im Transportmodus ausgeführt wird, besteht die Einschränkung, dass die IP-Quell- und Ziel-Adressen des zu verschlüsselnden Pakets mit den IPsec-Peer-Adressen (dem Router selbst) übereinstimmen müssen. In diesem Fall bedeutet dies lediglich, dass der GRE-Tunnel-Endpunkt und die IPsec-Peer-Adressen identisch sein müssen. Dies ist kein Problem, da es sich

bei den gleichen Routern sowohl um IPsec- als auch um GRE-Tunnel-Endpunkte handelt. Durch die Kombination von GRE-Tunneln mit IPsec-Verschlüsselung können Sie ein dynamisches IP-Routing-Protokoll verwenden, um die Routing-Tabellen an beiden Enden des verschlüsselten Tunnels zu aktualisieren. Die Einträge der IP-Routing-Tabelle für die Netzwerke, die durch den verschlüsselten Tunnel erfasst wurden, haben das andere Ende des Tunnels (IP-Adresse der GRE-Tunnelschnittstelle) als IP Next Hop. Wenn sich also die Netzwerke auf beiden Seiten des Tunnels ändern, wird die andere Seite dynamisch von der Änderung erfahren, und die Verbindung wird ohne Konfigurationsänderungen auf den Routern fortgesetzt.

Basiskonfiguration

Im Folgenden sehen Sie eine standardmäßige Point-to-Point-IPsec+GRE-Konfiguration. Danach gibt es eine Reihe von Konfigurationsbeispielen, in denen schrittweise spezifische Funktionen der DMVPN-Lösung hinzugefügt werden, um die verschiedenen Funktionen von DMVPN zu veranschaulichen. Jedes Beispiel baut auf den vorherigen Beispielen auf, um die Verwendung der DMVPN-Lösung in immer komplexeren Netzwerkdesigns zu veranschaulichen. Diese Beispielfolge kann als Vorlage für die Migration eines aktuellen IPsec+GRE-VPN zu einem DMVPN verwendet werden. Sie können die Migration jederzeit unterbrechen, wenn das jeweilige Konfigurationsbeispiel Ihren Anforderungen an das Netzwerkdesign entspricht.

IPsec + GRE Hub and Spoke (n = 1,2,3,...)



Hub-Router

```
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
```

Spoke1-Router

```
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-Router

```
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> Router

```
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.
```

In der obigen Konfiguration werden ACLs verwendet, um festzulegen, welcher Datenverkehr verschlüsselt wird. Sowohl auf den Hub- als auch auf den Spoke-Routern muss diese ACL nur mit den IP-Paketen des GRE-Tunnels übereinstimmen. Unabhängig davon, wie sich die Netzwerke an beiden Enden verändern, ändern sich die GRE IP-Tunnelpakete nicht. Diese ACL muss also nicht geändert werden.

Hinweis: Wenn Sie Cisco IOS-Softwareversionen vor 12.2(13)T verwenden, müssen Sie den Konfigurationsbefehl **crypto map vpnmap1** sowohl auf die GRE-Tunnelschnittstellen (Tunnel<x>) als auch auf die physische Schnittstelle (Ethernet0) anwenden. Bei Cisco IOS Version 12.2(13)T und höher wenden Sie nur den Konfigurationsbefehl **crypto map vpnmap1** auf die physische Schnittstelle (Ethernet0) an.

Beispiele für die Routing-Tabellen auf Hub-and-Spoke-Routern

Routing-Tabelle auf dem Hub-Router

```
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>
```

Routingtabelle auf Spoke1-Router

```
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
```

Routingtabelle auf Spoke<n> Router

```
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
```

Dies ist eine grundlegende Arbeitskonfiguration und dient als Ausgangspunkt für einen Vergleich mit den komplexeren Konfigurationen, die mit der DMVPN-Lösung möglich sind. Die erste Änderung reduziert die Größe der Konfiguration auf dem Hub-Router. Dies ist bei einer geringen Anzahl von Spoke-Routern nicht wichtig, ist jedoch von Bedeutung, wenn mehr als 50 bis 100 Spoke-Router vorhanden sind.

Reduzieren der Konfigurationsgröße für den Hub-Router

Im folgenden Beispiel wird die Konfiguration auf dem Hub-Router minimal von mehreren GRE-Point-to-Point-Tunnelschnittstellen auf eine einzelne GRE-Multipoint-Tunnelschnittstelle geändert. Dies ist ein erster Schritt zur DMVPN-Lösung.

Es gibt einen eindeutigen Block von Konfigurationslinien auf dem Hub-Router, um die Merkmale der Crypto Map für jeden Spoke-Router zu definieren. Dieser Teil der Konfiguration definiert die Krypto-ACL und die GRE-Tunnelschnittstelle für diesen Spoke-Router. Diese Eigenschaften sind meist für alle Spokes identisch, mit Ausnahme von IP-Adressen (**set peer ...**, **tunnel destination ...**).

Wenn Sie sich die obige Konfiguration auf den Hub-Router anschauen, sehen Sie, dass pro Spoke-Router mindestens 13 Konfigurationslinien vorhanden sind. vier für die Crypto Map, eine für die Crypto ACL und acht für die GRE-Tunnelschnittstelle. Die Gesamtzahl der Konfigurationsleitungen beträgt bei 300 Spoke-Routern 3.900 Zeilen. Sie benötigen außerdem 300 (/30) Subnetze, um die einzelnen Tunnelverbindungen zu adressieren. Eine Konfiguration dieser Größe ist sehr schwer zu verwalten und noch schwieriger bei der Fehlerbehebung im VPN-Netzwerk. Um diesen Wert zu reduzieren, können Sie dynamische Kryptokarten verwenden, die den obigen Wert um 1.200 Zeilen reduzieren würden, sodass 2.700 Zeilen in einem 300-Spoke-Netzwerk verbleiben.

Hinweis: Bei der Verwendung dynamischer Crypto Maps muss der IPsec-Verschlüsselungstunnel vom Spoke-Router initiiert werden. Sie können auch **ip unnumbered <interface>** verwenden, um die Anzahl der Subnetze zu reduzieren, die für die GRE-Tunnel benötigt werden. Dies kann die Fehlerbehebung jedoch später erschweren.

Mit der DMVPN-Lösung können Sie eine einzelne GRE-Tunnelschnittstelle für mehrere Punkte und ein einziges IPsec-Profil auf dem Hub-Router für die Verarbeitung aller Spoke-Router konfigurieren. Dadurch kann die Konfiguration auf dem Hub-Router konstant bleiben, unabhängig davon, wie viele Spoke-Router dem VPN-Netzwerk hinzugefügt werden.

Die DMVPN-Lösung führt die folgenden neuen Befehle ein:

```
crypto ipsec profile
```

Der Befehl **crypto ipsec profile <name>** wird wie eine dynamische Crypto Map verwendet und wurde speziell für Tunnelschnittstellen entwickelt. Mit diesem Befehl werden die Parameter für die IPsec-Verschlüsselung auf den Spoke-to-Hub- und den Spoke-to-Spoke-VPN-Tunneln definiert. Der einzige Parameter, der für das Profil erforderlich ist, ist der Transformationssatz. Die IPsec-Peer-Adresse und die **Match-Adresse ...** -Klausel für den IPsec-Proxy werden automatisch von den NHRP-Zuordnungen für den GRE-Tunnel abgeleitet.

Der Befehl **tunnel protection ipsec profile <name>** wird unter der GRE-Tunnelschnittstelle konfiguriert und wird verwendet, um die GRE-Tunnelschnittstelle dem IPsec-Profil zuzuordnen. Zusätzlich kann der Befehl **tunnel protection ipsec profile <name>** auch mit einem Point-to-Point GRE-Tunnel verwendet werden. In diesem Fall werden die IPsec-Peer- und Proxyinformationen von der **Tunnelquelle** abgeleitet **...** und **Tunnelziel ...** Konfiguration. Dies vereinfacht die Konfiguration, da der IPsec-Peer und die Krypto-ACLs nicht mehr benötigt werden.

Hinweis: Der Befehl **tunnel protection ...** gibt an, dass die IPsec-Verschlüsselung erfolgt, nachdem die GRE-Kapselung dem Paket hinzugefügt wurde.

Diese ersten beiden neuen Befehle ähneln der Konfiguration einer Crypto Map und der Zuweisung der Crypto Map zu einer Schnittstelle mithilfe des Befehls **crypto map <name>**. Der große Unterschied besteht darin, dass Sie mit den neuen Befehlen weder die IPsec-Peer-Adresse noch eine ACL angeben müssen, um die zu verschlüsselnden Pakete abzugleichen. Diese Parameter werden automatisch aus den NHRP-Zuordnungen für die mGRE-Tunnelschnittstelle bestimmt.

Hinweis: Bei Verwendung des **Tunnelschutzes ...** Befehl auf der Tunnel-Schnittstelle, eine **Kryptoübersicht ...** ist auf der physischen ausgehenden Schnittstelle nicht konfiguriert.

Der letzte neue Befehl, **ip nhrp map multicast dynamic**, ermöglicht dem NHRP das automatische Hinzufügen von Spoke-Routern zu den Multicast-NHRP-Zuordnungen, wenn diese Spoke-Router den mGRE+IPsec-Tunnel initiieren und ihre Unicast-NHRP-Zuordnungen registrieren. Dies ist erforderlich, damit dynamische Routing-Protokolle über die mGRE+IPsec-Tunnel zwischen Hub und Spokes arbeiten können. Wäre dieser Befehl nicht verfügbar, müsste der Hub-Router über eine separate Konfigurationsleitung für eine Multicast-Zuordnung zu den einzelnen Spokes verfügen.

Hinweis: Bei dieser Konfiguration müssen die Spoke-Router die mGRE+IPsec-Tunnelverbindung initiieren, da der Hub-Router nicht mit Informationen über die Stationen konfiguriert ist. Dies ist jedoch kein Problem, da bei DMVPN der mGRE+IPsec-Tunnel automatisch initiiert wird, wenn der Spoke-Router gestartet wird, und immer aktiv bleibt.

Hinweis: Das folgende Beispiel zeigt Point-to-Point-GRE-Tunnelschnittstellen auf den Spoke-Routern und die NHRP-Konfigurationslinien, die auf den Hub- und Spoke-Routern hinzugefügt wurden, um den mGRE-Tunnel auf dem Hub-Router zu unterstützen. Die Konfigurationsänderungen sind wie folgt:

Hub-Router (alt)

```
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
```

Hub-Router (neu)

```
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> Router (alt)

```
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 (neu)

```
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
 !
```

Auf den Spoke-Routern wurde die Subnetzmaske geändert, und unter der Tunnelschnittstelle

wurden NHRP-Befehle hinzugefügt. Die NHRP-Befehle sind erforderlich, da der Hub-Router jetzt NHRP verwendet, um die IP-Adresse der Spoke-Tunnel-Schnittstelle der IP-Adresse der Spoke-physischen Schnittstelle zuzuordnen.

```
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
```

Das Subnetz ist jetzt /24 anstelle von /30, daher befinden sich alle Knoten im gleichen Subnetz anstelle verschiedener Subnetze. Die Stationen senden weiterhin Spoke-to-Spoke-Datenverkehr über den Hub, da sie eine Point-to-Point-GRE-Tunnelschnittstelle verwenden. Die **ip nhrp-Authentifizierung ...**, **ip nhrp network-id ...** und **Tunnelschlüssel ...** werden verwendet, um die Tunnelpakete und die NHRP-Pakete der richtigen GRE-Tunnelschnittstelle und dem NHRP-Netzwerk zuzuordnen, wenn sie am Hub empfangen werden. Die **ip nhrp map ...** und **ip nhrp nhs** ... -Befehle verwendet NHRP in dem Spoke, um die NHRP-Zuordnung (10.0.0.<n+1> —> 172.16.<n>.1) zu dem Hub anzuzeigen. Die Adresse 10.0.0.<n+1> wird aus der **IP-Adresse** abgerufen **...** auf der Tunnelschnittstelle und die Adresse 172.16.<n>.1 wird vom **Tunnelziel** abgerufen **...** auf der Tunnelschnittstelle.

Wenn es 300 Spoke-Router gibt, würde durch diese Änderung die Anzahl der Konfigurationsleitungen auf dem Hub von 3900 auf 16 Leitungen reduziert (eine Verringerung um 3884 Leitungen). Die Konfiguration auf jedem Spoke-Router würde um 6 Leitungen erhöht.

Unterstützung dynamischer Adressen auf Spokes

Auf einem Cisco Router muss jeder IPsec-Peer mit der IP-Adresse des anderen IPsec-Peers konfiguriert werden, bevor der IPsec-Tunnel aktiviert werden kann. Dies kann problematisch sein, wenn ein Spoke-Router über eine dynamische Adresse auf seiner physischen Schnittstelle verfügt. Dies ist bei Routern üblich, die über DSL- oder Kabelverbindungen verbunden sind.

TED ermöglicht es einem IPsec-Peer, einen anderen IPsec-Peer zu finden, indem ein spezielles Internet Security Association and Key Management Protocol (ISAKMP)-Paket an die IP-Zieladresse des ursprünglichen Datenpakets gesendet wird, das verschlüsselt werden musste. Es wird davon ausgegangen, dass dieses Paket das intervenierende Netzwerk über denselben Pfad durchläuft wie das IPsec-Tunnelpaket. Dieses Paket wird vom anderen-End-IPsec-Peer übernommen, der auf den ersten Peer antwortet. Die beiden Router handeln dann ISAKMP und IPsec Security Associations (SAs) aus und starten den IPsec-Tunnel. Dies funktioniert nur, wenn die zu verschlüsselnden Datenpakete über routingfähige IP-Adressen verfügen.

TED kann in Kombination mit den GRE-Tunneln verwendet werden, wie im vorherigen Abschnitt konfiguriert. Dies wurde getestet und funktioniert, obwohl es einen Fehler in früheren Versionen

der Cisco IOS-Software gab, bei dem TED den gesamten IP-Datenverkehr zwischen den beiden IPsec-Peers zur Verschlüsselung zwang, nicht nur die GRE-Tunnelpakete. Die DMVPN-Lösung bietet diese und zusätzliche Funktionen, ohne dass die Hosts routingfähige IP-Adressen im Internet verwenden müssen und ohne Sonde- und Antwortpakete senden zu müssen. Mit einer geringfügigen Änderung kann die Konfiguration aus dem letzten Abschnitt zur Unterstützung von Spoke-Routern mit dynamischen IP-Adressen an ihren externen physischen Schnittstellen verwendet werden.

Hub-Router (keine Änderung)

```
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> Router (alt)

```
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.
```

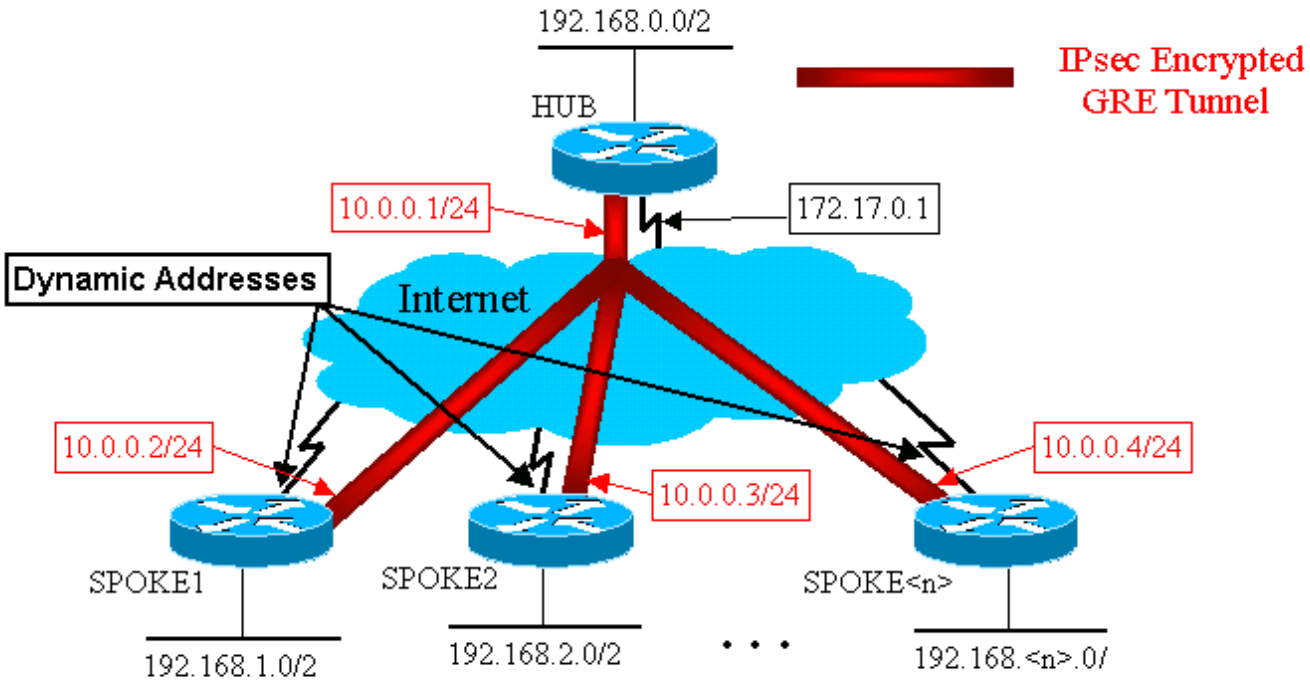
Spoke<n> Router (neu)

```
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
```

In der neuen Spoke-Konfiguration werden folgende Funktionen verwendet:

- Wenn die GRE-Tunnelschnittstelle aktiviert wird, sendet sie jetzt NHRP-Registrierungspakete an den Hub-Router. Diese NHRP-Registrierungspakete lösen die Initiierung von IPsec aus. Auf dem Spoke-Router werden die Befehle set peer <Peer-Adresse> und **match ip access-list <ACL>** konfiguriert. Die ACL gibt GRE als Protokoll, Any für die Quelle und die Hub-IP-Adresse für das Ziel an.**Hinweis:** Hierbei ist zu beachten, dass alle Router als Quelle in der ACL verwendet werden. Dies muss der Fall sein, da die IP-Adresse des Spoke-Routers dynamisch ist und daher vor dem Aktivieren der physischen Schnittstelle nicht bekannt ist. Ein IP-Subnetz kann für die Quelle in der ACL verwendet werden, wenn die Adresse der dynamischen Spoke-Schnittstelle auf eine Adresse innerhalb dieses Subnetzes beschränkt wird.
- Der Befehl **set security-zuordnung level per host** wird verwendet, sodass die IP-Quelle im Spokes-IPsec-Proxy nur die aktuelle physische Schnittstellenadresse (/32) und nicht die "any" aus der ACL ist. Wenn das "Any" von der ACL als Quelle im IPsec-Proxy verwendet würde, würde es verhindern, dass jeder andere Spoke-Router mit diesem Hub auch einen IPsec+GRE-Tunnel einrichten kann. Dies liegt daran, dass der resultierende IPsec-Proxy auf dem Hub der **Berechtigung "gre host 172.17.0.1 any"** entspricht. Dies würde bedeuten, dass alle GRE-Tunnelpakete, die für einen beliebigen Spoke bestimmt sind, verschlüsselt und an den ersten Spoke gesendet werden, der einen Tunnel mit dem Hub eingerichtet hat, da sein IPsec-Proxy GRE-Pakete für jeden Spoke vergleicht.
- Nach der Einrichtung des IPsec-Tunnels wird vom Spoke-Router ein NHRP-Registrierungspaket zum konfigurierten Next Hop Server (NHS) gesendet. Der NHS ist der Hub-Router dieses Hub-and-Spoke-Netzwerks. Das NHRP-Registrierungspaket enthält Informationen zum Erstellen einer NHRP-Zuordnung für diesen Spoke-Router durch den Hub-Router. Mit dieser Zuordnung kann der Hub-Router Unicast-IP-Datenpakete über den mGRE+IPsec-Tunnel an diesen Spoke-Router weiterleiten. Außerdem fügt der Hub den Spoke-Router seiner NHRP-Multicast-Zuordnungsliste hinzu. Der Hub sendet dann dynamische IP-Routing-Multicast-Pakete an den Spoke (wenn ein dynamisches Routing-Protokoll konfiguriert ist). Der Spoke-Router wird dann zum Routing-Protokoll-Nachbarn des Hubs und tauscht Routing-Updates aus.

IPsec + mGRE-Hub und Spoke



Hub-Router

```
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
!
```

Beachten Sie in der obigen Hub-Konfiguration, dass die IP-Adressen der Spoke-Router nicht konfiguriert sind. Die externe physische Schnittstelle des Spokes und die Zuordnung zu den Tunnelschnittstellen-IP-Adressen des Spokes werden vom Hub dynamisch über NHRP erfasst. Dadurch kann die externe IP-Adresse der physischen Schnittstelle des Spokes dynamisch zugewiesen werden.

Spoke1-Router

```
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-Router

```
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
```

Wichtigste Merkmale der Spoke-Konfigurationen:

- Die IP-Adresse der externen physischen Schnittstelle (Ethernet0) ist über DHCP dynamisch.**ip address dhcp hostname Spoke2**
- Die Crypto ACL (101) gibt ein Subnetz als Quelle für den IPsec-Proxy an.**access-list 101 permit gre 172.16.2.0 0.0.0.255 host 172.17.0.1**
- Der folgende Befehl in der IPsec-Crypto Map gibt an, dass die Sicherheitszuordnung pro Host erfolgt.**Sicherheitszuordnungsstufe pro Host festlegen**

- Alle Tunnel sind Teil desselben Subnetzes, da alle über dieselbe GRE-Multipoint-Schnittstelle am Hub-Router verbunden sind.**ip address 10.0.0.0.2 255.255.255.0**

Durch die Kombination dieser drei Befehle ist es nicht erforderlich, die IP-Adresse der externen physischen Schnittstelle des Spokes zu konfigurieren. Der verwendete IPsec-Proxy ist Host- und nicht Subnetzbasiert.

Für die Konfiguration der Spoke-Router ist die IP-Adresse des Hub-Routers konfiguriert, da der IPsec+GRE-Tunnel initiiert werden muss. Beachten Sie die Ähnlichkeit zwischen den Spoke1- und Spoke2-Konfigurationen. Diese beiden Konfigurationen sind nicht nur ähnlich, sondern alle Spoke-Router-Konfigurationen sind ähnlich. In den meisten Fällen benötigen alle Spokes lediglich eindeutige IP-Adressen auf ihren Schnittstellen, und die übrigen Konfigurationen sind identisch. So können viele Spoke-Router schnell konfiguriert und bereitgestellt werden.

Die NHRP-Daten sehen auf dem Hub and Spoke wie folgt aus.

Hub-Router

```
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-Router

```
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
```

Dynamische Multipoint-Hub-and-Spoke

Die Konfiguration auf den Spoke-Routern oben stützt sich nicht auf Funktionen der DMVPN-Lösung, sodass auf den Spoke-Routern Cisco IOS-Softwareversionen vor 12.2(13)T ausgeführt werden können. Die Konfiguration des Hub-Routers basiert auf DMVPN-Funktionen, daher muss Cisco IOS Version 12.2(13)T oder höher ausgeführt werden. Dadurch können Sie flexibel entscheiden, wann Sie ein Upgrade der bereits bereitgestellten Spoke-Router durchführen müssen. Wenn auf Ihren Spoke-Routern auch Cisco IOS Version 12.2(13)T oder höher ausgeführt wird, können Sie die Spoke-Konfiguration wie folgt vereinfachen.

Spoke<n> Router (vor 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 (nach 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>
 !
```

Beachten Sie, dass wir Folgendes durchgeführt haben:

1. Entfernen Sie den Befehl **crypto map vpnmap1 10 ipsec-isakmp** und ersetzten ihn durch

crypto ipsec profile vpnprof.

2. Den Befehl **crypto map vpnmap1** von den Ethernet0-Schnittstellen entfernt und den Befehl **tunnel protection ipsec profile vpnprof** auf die Tunnel0-Schnittstelle **platziert**.
3. Krypto-ACL entfernt, **access-list 101 permit gre für jeden Host 172.17.0.1**.

In diesem Fall werden die IPsec-Peer-Adressen und -Proxys automatisch von der **Tunnelquelle** abgeleitet **...** und **Tunnelziel ...** Konfiguration. Die Peers und Proxys sind wie folgt (wie in der Ausgabe von **show crypto ipsec als** Befehl gesehen):

```
...
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
...
```

Zusammenfassend lässt sich sagen, dass die folgenden Vollkonfigurationen alle bis zu diesem Zeitpunkt aus der Basiskonfiguration vorgenommenen Änderungen enthalten (IPsec+GRE-Hub and Spoke).

Hub-Router

```
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
 !
```

Die Hub-Konfiguration ändert sich nicht.

Spoke1-Router

```
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-Router

```
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
!
```

Dynamisches Multipoint-IPsec-VPN

Die Konzepte und Konfigurationen in diesem Abschnitt zeigen alle Funktionen von DMVPN. NHRP ermöglicht es den Spoke-Routern, die externen physischen Schnittstellenadressen der anderen Spoke-Router im VPN-Netzwerk dynamisch zu erfassen. Das bedeutet, dass ein Spoke-Router über genügend Informationen verfügt, um einen IPsec+mGRE-Tunnel dynamisch direkt zu anderen Spoke-Routern zu erstellen. Dies ist von Vorteil, da dieser Spoke-to-Spoke-Datenverkehr, der über den Hub-Router gesendet wird, verschlüsselt/entschlüsselt werden muss, wodurch die Verzögerung und die Last auf dem Hub-Router zweimal erhöht werden. Um diese Funktion nutzen zu können, müssen die Spoke-Router von Point-to-Point GRE (p-pGRE)- zu Multipoint GRE (mGRE)-Tunnelschnittstellen geschaltet werden. Außerdem müssen sie sich mit den (Unter-)Netzwerken vertraut machen, die hinter den anderen Stationen mit einem IP Next-Hop der Tunnel-IP-Adresse des anderen Spoke-Routers verfügbar sind. Die Spoke-Router erlernen diese (Sub-)Netzwerke über das dynamische IP-Routing-Protokoll, das über den IPsec+mGRE-Tunnel mit dem Hub ausgeführt wird.

Das dynamische IP-Routing-Protokoll, das auf dem Hub-Router ausgeführt wird, kann so konfiguriert werden, dass die von einem Spoke-System an dieselbe Schnittstelle zurückgegebenen Routen zu allen anderen Spokes wiedergegeben werden. Der IP Next-Hop auf diesen Routen ist jedoch in der Regel der Hub-Router und nicht der Spoke-Router, von dem der Hub diese Route gelernt hat.

Hinweis: Das dynamische Routing-Protokoll wird nur auf den Hub-and-Spoke-Verbindungen ausgeführt, nicht jedoch auf den dynamischen Spoke-to-Spoke-Verbindungen.

Die dynamischen Routing-Protokolle (RIP, OSPF und EIGRP) müssen auf dem Hub-Router konfiguriert werden, um die Routen an der mGRE-Tunnelschnittstelle anzuzeigen und den IP Next-Hop auf den ursprünglichen Spoke-Router für Routen festzulegen, die von einem Spoke gelernt wurden, wenn die Route an die anderen Spokes zurückgegeben wird.

Die folgenden Anforderungen gelten für die Routing-Protokoll-Konfigurationen:

RIP

Sie müssen Split Horizon auf der mGRE-Tunnelschnittstelle am Hub deaktivieren, da RIP ansonsten keine Routen ankündigen kann, die über die mGRE-Schnittstelle an derselben Schnittstelle erfasst wurden.

```
no ip split-horizon
```

Weitere Änderungen sind nicht erforderlich. RIP verwendet automatisch den ursprünglichen IP-Next-Hop auf Routen, die über dieselbe Schnittstelle gemeldet werden, über die die Routen abgerufen wurden.

EIGRP

Sie müssen Split Horizon auf der mGRE-Tunnelschnittstelle am Hub deaktivieren, da EIGRP ansonsten keine Routen ankündigt, die über die mGRE-Schnittstelle an derselben Schnittstelle erfasst wurden.

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

EIGRP legt standardmäßig fest, dass der IP Next-Hop der Hub-Router für Routen ist, die er anzeigt, selbst wenn diese Routen über dieselbe Schnittstelle zurückgemeldet werden, über die er sie gelernt hat. In diesem Fall benötigen Sie den folgenden Konfigurationsbefehl, um EIGRP anzuweisen, beim Anzeigen dieser Routen den ursprünglichen IP Next-Hop zu verwenden.

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

Hinweis: Der Befehl **no ip next-hop-self eigrp <as>** steht ab Cisco IOS Release 12.3(2) zur Verfügung. Für Cisco IOS-Versionen zwischen 12.2(13)T und 12.3(2) müssen Sie folgende Schritte ausführen:

- Wenn dynamische Spoke-to-Spoke-Tunnel nicht benötigt werden, ist der obige Befehl nicht erforderlich.
- Wenn dynamische Spoke-to-Spoke-Tunnel benötigt werden, müssen Sie das Switching auf der Tunnelschnittstelle der Spoke-Router durchführen.
- Andernfalls müssen Sie ein anderes Routing-Protokoll über das DMVPN verwenden.

OSPF

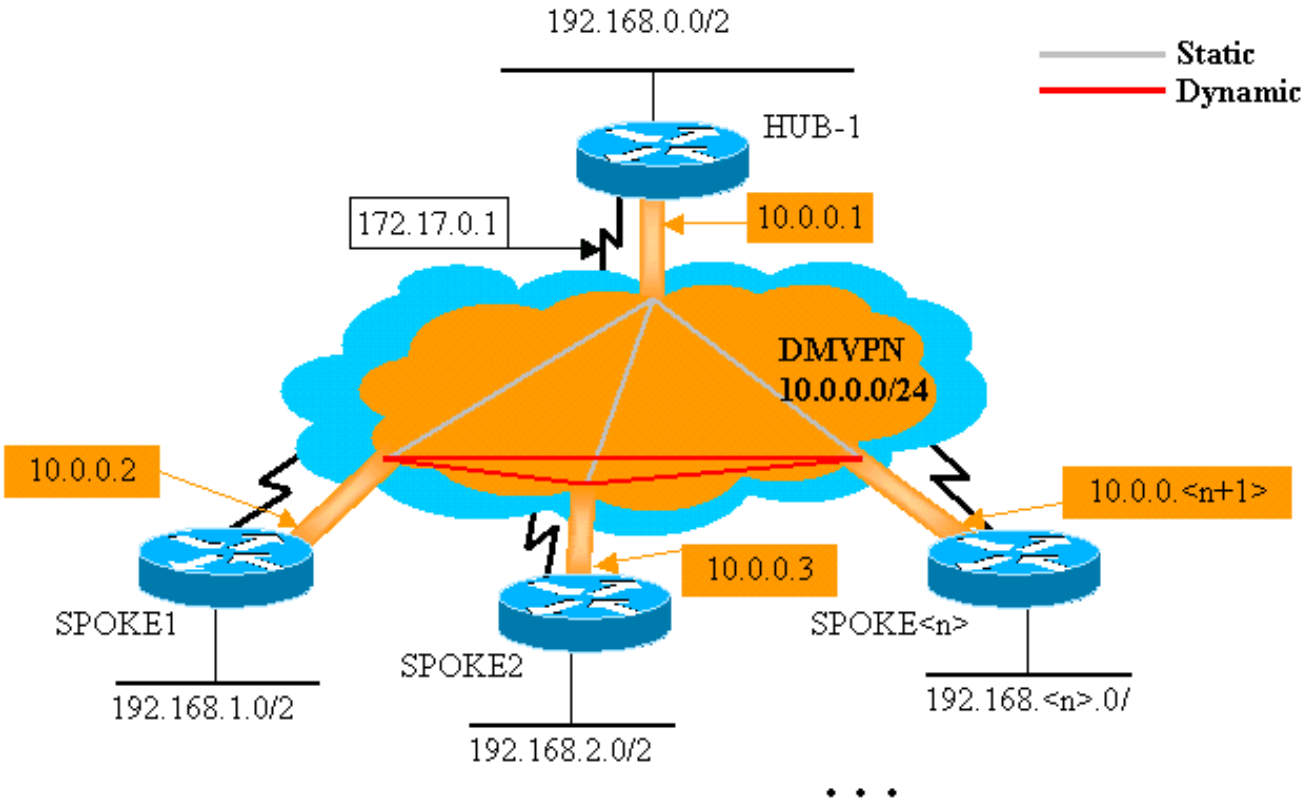
Da OSPF ein Link-State-Routing-Protokoll ist, gibt es keine Split Horizon-Probleme. In der Regel konfigurieren Sie für Multipoint-Schnittstellen den OSPF-Netzwerktyp als Point-to-Multipoint, dies führt jedoch dazu, dass OSPF der Routing-Tabelle der Spoke-Router Host-Routen hinzufügt. Diese Hostrouten würden dazu führen, dass Pakete, die an Netzwerke hinter anderen Spoke-Routern gerichtet sind, über den Hub weitergeleitet und dann direkt an den anderen Spoke weitergeleitet werden. Um dieses Problem zu umgehen, konfigurieren Sie den OSPF-Netzwerktyp für die Übertragung mit dem Befehl.

```
ip ospf network broadcast
```

Sie müssen außerdem sicherstellen, dass der Hub-Router der designierte Router (DR) für das IPsec+mGRE-Netzwerk ist. Dies wird erreicht, indem die OSPF-Priorität auf dem Hub auf größer als 1 und auf den Stationen auf 0 festgelegt wird.

- Hub: **ip ospf Priorität 2**
- Spoke: **ip ospf priority 0**

DMVPN-Single-Hub



Hub-Router

```
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
 !
```

Die einzige Änderung in der Hub-Konfiguration besteht darin, dass OSPF das Routing-Protokoll anstelle von EIGRP ist. Beachten Sie, dass der OSPF-Netzwerktyp auf "Senden" und die Priorität auf "2" festgelegt ist. Wenn der OSPF-Netzwerktyp für Broadcast festgelegt wird, installiert OSPF Routen für Netzwerke hinter den Spokes-Routern mit einer IP Next-Hop-Adresse als GRE-Tunneladresse für diesen Spoke-Router.

Spoke1-Router

```
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
!
```

Die Konfiguration auf den Spoke-Routern ähnelt nun der Konfiguration auf dem Hub. Die Unterschiede sind:

- Die OSPF-Priorität ist auf 0 festgelegt. Es ist nicht zulässig, dass die Spoke-Router zum DR für das mGRE-Nonbroadcast Multiaccess (NBMA)-Netzwerk werden. Nur der Hub-Router verfügt über direkte statische Verbindungen zu allen Spoke-Routern. Der DR muss Zugang zu allen Mitgliedern des NBMA-Netzwerks haben.
- Für den Hub-Router sind NHRP-Unicast- und Multicast-Zuordnungen konfiguriert.

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

  In der vorherigen Konfiguration wurde Multicast über die **ip nhrp-Karte zugeordnet ...** -Befehl nicht benötigt, da der GRE-Tunnel Point-to-Point war. In diesem Fall werden die Multicast-Pakete automatisch über den Tunnel an ein einziges mögliches Ziel gekapselt. Dieser Befehl wird jetzt benötigt, da der Stsprecher-GRE-Tunnel zu Multipoint geändert wurde und es mehr als ein mögliches Ziel gibt.
- Wenn der Spoke-Router aktiviert wird, muss er die Tunnelverbindung mit dem Hub initiieren, da der Hub-Router nicht mit Informationen über die Spoke-Router konfiguriert ist und die Spoke-Router möglicherweise dynamisch IP-Adressen zugewiesen haben. Die Spoke-Router sind auch mit dem Hub als NHRP NHS konfiguriert.
  ```
  ip nhrp nhs 10.0.0.1
  ```

Mit dem obigen Befehl sendet der Spoke-Router in regelmäßigen Abständen NHRP-Registrierungspakete über den mGRE+IPsec-Tunnel an den Hub-Router. Diese Registrierungspakete stellen die Spoke-NHRP-Zuordnungsinformationen bereit, die der Hub-Router zum Tunnel von Paketen zu den Spoke-Routern benötigt.

Spoke2-Router

```
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> Router

```
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.




 !
```

Beachten Sie, dass die Konfigurationen aller Spoke-Router sehr ähnlich sind. Die einzigen Unterschiede sind die IP-Adressen an den lokalen Schnittstellen. Dies hilft bei der Bereitstellung einer großen Anzahl von Spoke-Routern. Alle Spoke-Router können identisch konfiguriert werden, und nur die lokalen IP-Schnittstellenadressen müssen hinzugefügt werden.

Sehen Sie sich an diesem Punkt die Routing-Tabellen und die NHRP-Zuordnungstabellen der Router Hub, Spoke1 und Spoke2 an, um die Ausgangsbedingungen (kurz nachdem die Router Spoke1 und Spoke2 hochgefahren wurden) und die Bedingungen zu sehen, unter denen Spoke1 und Spoke2 eine dynamische Verbindung zwischen ihnen hergestellt haben.

Anfängliche Bedingungen

Informationen zum Hub-Router

```
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-Routerinformationen

```
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-Router-Informationen

```
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
```

An diesem Punkt pingen wir von 192.168.1.2 bis 192.168.2.3. Diese Adressen gelten jeweils für Hosts hinter den Spoke1- bzw. Spoke2-Routern. Die folgende Ereignissequenz wird beim Aufbau des direkten Spoke-to-Spoke-mGRE+IPsec-Tunnels angewendet.

1. Der Spoke1-Router empfängt das Ping-Paket mit dem Ziel 192.168.2.3. Es sucht dieses Ziel in der Routing-Tabelle und stellt fest, dass es dieses Paket über die Tunnel0-Schnittstelle an die IP-Next-Hop 10.0.0.3 weiterleiten muss.
2. Der Spoke1-Router überprüft die NHRP-Zuordnungstabelle für das Ziel 10.0.0.3 und stellt fest, dass kein Eintrag vorhanden ist. Der Spoke1-Router erstellt ein NHRP-Auflösungsanforderungspaket und sendet es an seinen NHS (den Hub-Router).
3. Der Hub-Router überprüft seine NHRP-Zuordnungstabelle für das Ziel 10.0.0.3 und stellt fest, dass er der Adresse 172.16.2.75 zugeordnet ist. Der Hub-Router erstellt ein Antwortpaket mit NHRP-Auflösung und sendet es an den Spoke1-Router.
4. Der Spoke1-Router empfängt die NHRP-Auflösungsantwort und gibt die 10.0.0.3 ->172.16.2.75-Zuordnung in seine NHRP-Zuordnungstabelle ein. Durch Hinzufügen der NHRP-Zuordnung wird IPsec veranlasst, einen IPsec-Tunnel mit dem Peer 172.16.2.75 zu initiieren.
5. Der Spoke1-Router initiiert ISAKMP mit 172.16.2.75 und handelt die ISAKMP- und IPsec-SAs aus. Der IPsec-Proxy wird aus dem Tunnel0-**Tunnelquelle <address>**-Befehl und der NHRP-Zuordnung abgeleitet.

   ```
   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. Nachdem der IPsec-Tunnel fertig erstellt wurde, werden alle weiteren Datenpakete zum Subnetz 192.168.2.0/24 direkt an Spoke2 gesendet.
7. Nachdem ein Paket für 192.168.2.3 an den Host weitergeleitet wurde, sendet dieser Host ein Rückgabepaket an 192.168.1.2. Wenn der Spoke2-Router dieses Paket für 192.168.1.2 empfängt, sucht er dieses Ziel in der Routing-Tabelle und stellt fest, dass es dieses Paket an die Tunnel0-Schnittstelle an den IP Next-Hop 10.0.0.2 weiterleiten muss.
8. Der Spoke2-Router überprüft die NHRP-Zuordnungstabelle für das Ziel 10.0.0.2 und stellt

fest, dass kein Eintrag vorhanden ist. Der Spoke2-Router erstellt ein NHRP-Lösungsanforderungspaket und sendet es an seinen NHS (den Hub-Router).

9. Der Hub-Router überprüft seine NHRP-Zuordnungstabelle für das Ziel 10.0.0.2 und stellt fest, dass er der Adresse 172.16.1.24 zugeordnet ist. Der Hub-Router erstellt ein Antwortpaket mit NHRP-Auflösung und sendet es an den Spoke2-Router.
10. Der Spoke2-Router empfängt die NHRP-Auflösungsantwort und gibt die 10.0.0.2 -> 172.16.1.24-Zuordnung in seine NHRP-Zuordnungstabelle ein. Durch Hinzufügen der NHRP-Zuordnung wird IPsec veranlasst, einen IPsec-Tunnel mit dem Peer 172.16.1.24 zu initiieren. Es gibt jedoch bereits einen IPsec-Tunnel mit Peer 172.16.1.24, daher muss nichts weiter getan werden.
11. Spoke1 und Spoke2 können jetzt Pakete direkt miteinander weiterleiten. Wenn die NHRP-Zuordnung nicht für die Weiterleitung von Paketen für die Haltezeit verwendet wurde, wird die NHRP-Zuordnung gelöscht. Durch das Löschen des NHRP-Zuordnungseintrags wird IPsec veranlasst, die IPsec-SAs für diese direkte Verbindung zu löschen.

Bedingungen nach dem Erstellen einer dynamischen Verbindung zwischen Spoke1 und Spoke2

Spoke1-Routerinformationen

```
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-Router-Informationen

```
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
```

Aus der obigen Ausgabe können Sie sehen, dass Spoke1 und Spoke2 NHRP-Zuordnungen voneinander vom Hub-Router erhalten haben und einen mGRE+IPsec-Tunnel erstellt und verwendet haben. Die NHRP-Zuordnungen laufen nach fünf Minuten ab (der aktuelle Wert der NHRP-Holdtime = 300 Sekunden). Wenn die NHRP-Zuordnungen innerhalb der letzten Minute vor dem Ablauf verwendet werden, werden eine NHRP-Auflösungsanfrage und eine Antwort gesendet, um den Eintrag zu aktualisieren, bevor er gelöscht wird. Andernfalls wird die NHRP-Zuordnung gelöscht, und IPsec löscht die IPsec-SAs.

Dynamisches Multipoint-IPsec-VPN mit zwei Hubs

Mit einigen zusätzlichen Konfigurationslinien für die Spoke-Router können Sie aus Redundanzgründen duale (oder mehrere) Hub-Router einrichten. Es gibt zwei Möglichkeiten zur Konfiguration von DMVPNs mit dualem Hub.

- Ein DMVPN-Netzwerk mit jeweils einer Spoke-Schnittstelle für einen Multipoint-GRE-Tunnel, das auf zwei verschiedene Hubs als Next-Hop-Server (NHS) verweist.Die Hub-Router verfügen nur über eine einzige GRE-Tunnelschnittstelle mit mehreren Punkten.
- Zwei DMVPN-Netzwerke mit jeweils zwei Spoke-Routern haben zwei GRE-Tunnelschnittstellen (entweder Point-to-Point oder Multipoint) und jeden GRE-Tunnel mit einem anderen Hub-Router verbunden.Auch hier verfügen die Hub-Router nur über eine einzige GRE-Tunnelschnittstelle für mehrere Punkte.

In den folgenden Beispielen wird die Konfiguration dieser beiden unterschiedlichen Szenarien für DMVPNs mit zwei Hub erläutert. In beiden Fällen unterscheiden sich die markierten Unterschiede in Bezug auf die DMVPN-Konfiguration mit einem Hub.

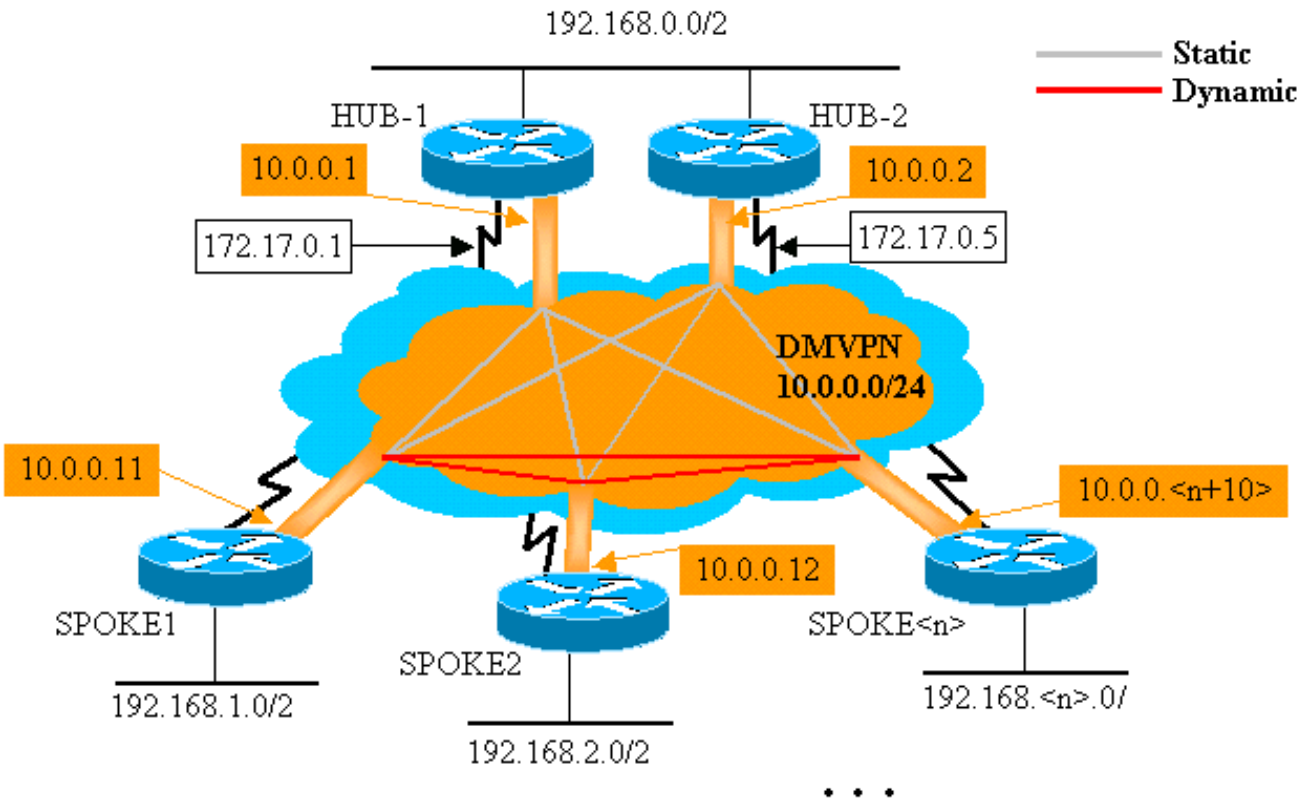
Dual-Hub - Einzel-DMVPN-Layout

Der duale Hub mit einem DMVPN-Layout ist relativ einfach einzurichten, bietet Ihnen aber nicht so viel Kontrolle über das Routing über das DMVPN wie der duale Hub mit dualem DMVPN-Layout. In diesem Fall soll eine einzige DMVPN-Cloud mit allen Hubs (in diesem Fall zwei) und allen Stationen, die mit diesem einzelnen Subnetz verbunden sind ("Cloud"), eingerichtet werden. Die statischen NHRP-Zuordnungen von den Stationen zu den Hubs definieren die statischen IPsec+mGRE-Verbindungen, über die das dynamische Routing-Protokoll ausgeführt wird. Das dynamische Routing-Protokoll wird nicht über die dynamischen IPsec+mGRE-Verbindungen zwischen Stationen ausgeführt. Da die Spoke-Router Nachbarn mit den Hub-Routern über

dieselbe mGRE-Tunnelschnittstelle weiterleiten, können Sie Link- oder Schnittstellenunterschiede (wie Metrik, Kosten, Verzögerung oder Bandbreite) nicht verwenden, um die Kennzahlen für das dynamische Routing-Protokoll zu ändern, um einen Hub dem anderen Hub vorzuziehen, wenn beide Verbindungen bestehen. Wenn diese Präferenz erforderlich ist, müssen interne Techniken für die Konfiguration des Routing-Protokolls verwendet werden. Aus diesem Grund ist es möglicherweise besser, EIGRP oder RIP anstelle von OSPF für das dynamische Routing-Protokoll zu verwenden.

Hinweis: Das oben genannte Problem tritt in der Regel nur dann auf, wenn sich die Hub-Router am gleichen Standort befinden. Wenn sie nicht am gleichen Standort arbeiten, bevorzugt das normale dynamische Routing den richtigen Hub-Router, selbst wenn das Zielnetzwerk über einen der Hub-Router erreicht werden kann.

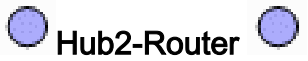
Dual-Hub - Einzel-DMVPN-Layout



Hub-Router

```
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
!
```




```
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
!
```

Die einzige Änderung in der Hub1-Konfiguration besteht darin, OSPF in zwei Bereiche zu ändern. Bereich 0 wird für das Netzwerk hinter den beiden Hubs und Bereich 1 für das DMVPN-Netzwerk und die Netzwerke hinter den Spoke-Routern verwendet. OSPF konnte einen Bereich verwenden, aber hier wurden zwei Bereiche verwendet, um die Konfiguration für mehrere OSPF-Bereiche zu demonstrieren.

Die Konfiguration für Hub2 entspricht im Prinzip der Hub1-Konfiguration mit den entsprechenden IP-Adressänderungen. Der Hauptunterschied besteht darin, dass Hub2 auch ein Spoke (oder Client) von Hub1 ist, wodurch Hub1 der primäre Hub und Hub2 der sekundäre Hub wird. Dies geschieht, sodass Hub2 ein OSPF-Nachbar mit Hub1 über den mGRE-Tunnel ist. Da Hub1 der OSPF-DR ist, muss er über eine direkte Verbindung mit allen anderen OSPF-Routern über die mGRE-Schnittstelle (NBMA-Netzwerk) verfügen. Ohne die direkte Verbindung zwischen Hub1 und Hub2 würde Hub2 nicht am OSPF-Routing teilnehmen, wenn auch Hub1 aktiv ist. Wenn Hub1 ausgefallen ist, ist Hub2 der OSPF-DR für das DMVPN (NBMA-Netzwerk). Wenn Hub1 wieder verfügbar ist, übernimmt Hub1 die OSPF-DR für das DMVPN.

Die Router hinter Hub1 und Hub2 verwenden Hub1 zum Senden von Paketen an die Spoke-Netzwerke, da die Bandbreite für die GRE-Tunnelschnittstelle auf 1000 Kb/s im Vergleich zu 900 Kbit/s auf Hub2 festgelegt ist. Im Gegensatz dazu senden die Spoke-Router Pakete für die Netzwerke hinter den Hub-Routern an Hub1 und Hub2, da auf jedem Spoke-Router nur eine mGRE-Tunnelschnittstelle vorhanden ist und es zwei Routen mit gleichen Kosten gibt. Wenn ein Load Balancing pro Paket verwendet wird, kann dies Pakete außerhalb der Reihenfolge verursachen.

Spoke1-Router

```
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
!
```

Die Konfiguration der Spoke-Router unterscheidet sich wie folgt:

- In der neuen Konfiguration wird die Spoke-Funktion mit statischen NHRP-Zuordnungen für Hub2 konfiguriert, und Hub2 wird als nächster Hop-Server hinzugefügt.Original:

  ```
  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
  ```

  Neu:

  ```
  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
  ```

- Die OSPF-Bereiche auf den Spoke-Routern wurden in Bereich 1 geändert.

Denken Sie daran, dass Sie das dynamische Routing-Protokoll über diesen Tunnel ausführen, indem Sie die statische NHRP-Zuordnung und den NHS auf einem Spoke-Router für einen Hub definieren. Dies definiert das Hub-and-Spoke-Routing oder das Nachbarnetzwerk. Beachten Sie, dass Hub2 ein Hub für alle Spokes und auch ein Spoke für Hub1 ist. So können bei Verwendung der DMVPN-Lösung mühelos Multilayer-Hub-and-Spoke-Netzwerke entworfen, konfiguriert und modifiziert werden.

Spoke2-Router

```
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> Router

```
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.




!
```

An dieser Stelle können Sie sich die Routing-Tabellen, die NHRP-Zuordnungstabellen und die IPsec-Verbindungen auf den Routern Hub1, Hub2, Spoke1 und Spoke2 ansehen, um die Ausgangsbedingungen (kurz nachdem die Spoke1- und Spoke2-Router hochgefahren wurden) anzuzeigen.

Anfängliche Bedingungen und Änderungen

Informationen zum Hub1-Router

```
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
```

Informationen zum Hub2-Router

```
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-Routerinformationen

```
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-Router-Informationen

```
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
```

Bei den Routing-Tabellen auf Hub1, Hub2, Spoke1 und Spoke2 gibt es einige interessante Probleme:

- Beide Hub-Router verfügen über Routen zu den Netzwerken hinter den Spoke-Routern zu gleichen Kosten.Hub 1:
  ```
  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
  ```
  Hub 2:
  ```
  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
  ```
  Das bedeutet, dass Hub1 und Hub2 den Routern im Netzwerk hinter den Hub-Routern dieselben Kosten für die Netzwerke hinter den Spoke-Routern melden. Beispielsweise sieht die Routing-Tabelle auf einem Router, R2, der direkt mit dem LAN 192.168.0.0/24 verbunden ist, wie folgt aus: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
  ```
- Die Spoke-Router verfügen hinter den Hub-Routern über Routen zu gleichen Kosten wie die Hub-Router zum Netzwerk.Spoke 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
  ```
  Spoke2:
  ```
  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
  ```
  Wenn die Spoke-Router einen Lastenausgleich pro Paket durchführen, können Pakete

außerhalb der Reihenfolge empfangen werden.

Um ein asymmetrisches Routing oder einen Lastenausgleich pro Paket über die Verbindungen zu den beiden Hubs zu vermeiden, müssen Sie das Routing-Protokoll so konfigurieren, dass es einen Spoke-to-Hub-Pfad in beide Richtungen bevorzugt. Wenn Sie möchten, dass Hub1 der primäre und Hub2 der Backup ist, können Sie die OSPF-Kosten für die Hub-Tunnel-Schnittstellen auf andere Werte einstellen.

Hub 1:

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

Hub 2:

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

Jetzt sehen die Routen wie folgt aus:

Hub 1:

```
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
```

Hub 2:

```
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
```

Die Kosten für die Routen der Netzwerke hinter den Spoke-Routern für die beiden Hub-Router sind jetzt unterschiedlich. Dies bedeutet, dass Hub1 für die Weiterleitung von Datenverkehr an die Spoke-Router bevorzugt wird, wie auf Router R2 zu erkennen ist. Dadurch wird das im ersten Aufzählungspunkt oben beschriebene asymmetrische Routing-Problem behoben.

Das asymmetrische Routing in die andere Richtung, wie im zweiten Aufzählungspunkt oben beschrieben, ist noch vorhanden. Wenn Sie OSPF als dynamisches Routing-Protokoll verwenden, können Sie dies mithilfe der **Distanz** beheben**.** unter **Router OSPF 1** auf den Stationen die über Hub1 abgefragten Routen gegenüber den über Hub2 empfangenen Routen bevorzugen.

Spoke 1:

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

Spoke2:

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

Jetzt sehen die Routen wie folgt aus:

Spoke 1:

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

Spoke2:

```
O     192.168.1.0/24 [110/11] via 10.0.0.1, 00:00:10, Tunnel0
```

Die obige Routing-Konfiguration schützt vor asymmetrischem Routing und ermöglicht gleichzeitig ein Failover auf Hub2, wenn Hub1 ausfällt. Dies bedeutet, dass bei beiden Hubs nur Hub1 verwendet wird. Wenn Sie beide Hubs verwenden möchten, indem Sie die Stationen über die Hubs hinweg ausbalancieren, mit Failover-Schutz und ohne asymmetrisches Routing, dann kann die Routing-Konfiguration komplex werden, insbesondere bei Verwendung von OSPF. Aus diesem Grund ist der folgende duale Hub mit Dual-DMVPN-Layout möglicherweise besser geeignet.

Dual-Hub - Dual-DMVPN-Layout

Der duale Hub mit dualem DMVPN-Layout ist etwas schwieriger einzurichten, bietet Ihnen aber eine bessere Kontrolle des Routings über das DMVPN. Es sollen zwei separate DMVPN-"Clouds" vorhanden sein. Jeder Hub (in diesem Fall zwei) ist mit einem DMVPN-Subnetz ("Cloud") verbunden, und die Stationen sind mit beiden DMVPN-Subnetzen ("Clouds") verbunden. Da die Spoke-Router Nachbarn mit beiden Hub-Routern über die beiden GRE-Tunnelschnittstellen weiterleiten, können Sie Schnittstellenkonfigurationsunterschiede (z. B. Bandbreite, Kosten und Verzögerung) verwenden, um die Kennzahlen für das dynamische Routing-Protokoll so zu ändern, dass ein Hub dem anderen Hub vorgezogen wird, wenn beide Verbindungen bestehen.

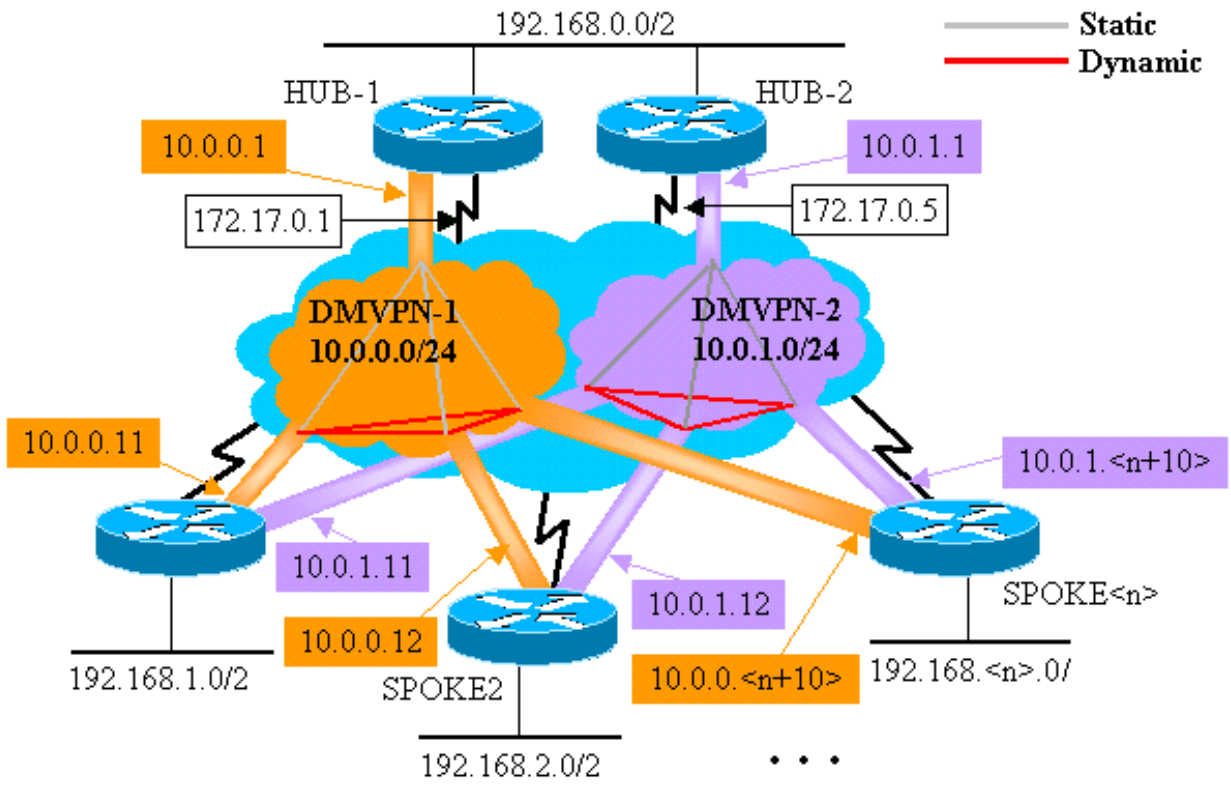
Hinweis: Das oben genannte Problem ist in der Regel nur relevant, wenn sich die Hub-Router am gleichen Standort befinden. Wenn sie nicht am gleichen Standort arbeiten, bevorzugt das normale dynamische Routing den richtigen Hub-Router, selbst wenn das Zielnetzwerk über einen der Hub-Router erreicht werden kann.

Sie können auf den Spoke-Routern entweder p-pGRE- oder mGRE-Tunnelschnittstellen verwenden. Mehrere p-pGRE-Schnittstellen auf einem Spoke-Router können die gleiche **Tunnelquelle** verwenden **...** IP-Adresse, aber mehrere mGRE-Schnittstellen auf einem Spoke-Router müssen eine eindeutige **Tunnelquelle** haben **...** IP-Adresse Das liegt daran, dass das erste Paket bei der Initiierung von IPsec ein ISAKMP-Paket ist, das einem der mGRE-Tunnel zugeordnet werden muss. Das ISAKMP-Paket hat nur die Ziel-IP-Adresse (Remote-IPsec-Peer-Adresse), mit der diese Zuordnung vorgenommen werden soll. Diese Adresse wird mit der **Tunnelquelle** abgeglichen **...** Adresse, aber da beide Tunnel die gleiche **Tunnelquelle** haben **...** - Adresse, wird immer die erste mGRE-Tunnelschnittstelle zugeordnet. Dies bedeutet, dass eingehende Multicast-Datenpakete möglicherweise der falschen mGRE-Schnittstelle zugeordnet werden, wodurch jedes dynamische Routing-Protokoll verletzt wird.

GRE-Pakete selbst haben dieses Problem nicht, da sie den **Tunnel-Schlüssel** haben **...** Wert, um zwischen den beiden mGRE-Schnittstellen zu unterscheiden. Ab den Cisco IOS Software-Versionen 12.3(5) und 12.3(7)T wurde ein zusätzlicher Parameter eingeführt, um diese

Einschränkung zu überwinden: **Tunnelschutz...shared**. Das **shared** Schlüsselwort gibt an, dass mehrere mGRE-Schnittstellen die IPSec-Verschlüsselung mit derselben Quell-IP-Adresse verwenden. Wenn Sie eine frühere Version haben, können Sie in diesem dualen Hub mit dualem DMVPN-Layout p-GRE-Tunnel verwenden. Im Fall des p-pGRE-Tunnels, beide **Tunnelquelle ...** und das **Tunnelziel ...** IP-Adressen können für die Zuordnung verwendet werden. In diesem Beispiel werden p-pGRE-Tunnel in diesem dualen Hub mit dualem DMVPN-Layout verwendet, jedoch nicht den **Shared** Qualifier.

Dual-Hub - Dual-DMVPN-Layout



Die folgenden hervorgehobenen Änderungen beziehen sich auf die zuvor in diesem Dokument dargestellten dynamischen Hub-and-Spoke-Konfigurationen für Multipoint-Verbindungen.

Hub1-Router

```
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-Router

```
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 diesem Fall sind die Hub1- und Hub2-Konfigurationen ähnlich. Der Hauptunterschied besteht darin, dass jeder der Hub eines anderen DMVPNs ist. Jedes DMVPN verwendet eine andere:

- IP-Subnetz (10.0.0.0/24, 10.0.0.1/24)
- NHRP-Netzwerk-ID (100000, 100001)
- Tunnel-Schlüssel (100000, 100001)

Das dynamische Routing-Protokoll wurde von OSPF auf EIGRP umgestellt, da die Einrichtung und Verwaltung eines NBMA-Netzwerks mit EIGRP einfacher ist, wie weiter unten in diesem Dokument beschrieben.

Spoke1-Router

```
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
!
```

Jeder der Spoke-Router ist mit zwei p-pGRE-Tunnelschnittstellen konfiguriert, einer in jedem der beiden DMVPNs. Die **IP-Adresse ...**, **ip nhrp network-id ...**, **Tunnelschlüssel ...** und **Tunnelziel ...** werden zur Unterscheidung zwischen den beiden Tunneln verwendet. Das dynamische Routing-Protokoll EIGRP wird über beide p-pGRE-Tunnel-Subnetze ausgeführt und dient der Auswahl einer p-pGRE-Schnittstelle (DMVPN) gegenüber der anderen.

Spoke2-Router

```
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> Router

```
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
!
```

Sehen wir uns an diesem Punkt die Routing-Tabellen, die NHRP-Zuordnungstabellen und die IPsec-Verbindungen auf den Routern Hub1, Hub2, Spoke1 und Spoke2 an, um die Ausgangsbedingungen (kurz nachdem die Spoke1- und Spoke2-Router hochgefahren wurden) anzuzeigen.

Anfängliche Bedingungen und Änderungen

Informationen zum Hub1-Router

```
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
```

Informationen zum Hub2-Router

```
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-Routerinformationen

```
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-Router-Informationen

```
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
```

Auch hier sind einige interessante Dinge über die Routing-Tabellen auf Hub1, Hub2, Spoke1 und Spoke2 zu beachten:

- Beide Hub-Router verfügen über Routen zu den Netzwerken hinter den Spoke-Routern zu gleichen Kosten.Hub 1:
  ```
  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
  ```
  Hub 2:
  ```
  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
  ```
  Das bedeutet, dass Hub1 und Hub2 den Routern im Netzwerk hinter den Hub-Routern dieselben Kosten für die Netzwerke hinter den Spoke-Routern melden. Beispielsweise sieht die Routing-Tabelle auf einem Router, R2, der direkt mit dem LAN 192.168.0.0/24 verbunden ist, wie folgt aus: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
  ```
- Die Spoke-Router verfügen hinter den Hub-Routern über Routen zu gleichen Kosten wie die Hub-Router zum Netzwerk.Spoke 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
  ```
  Spoke2:
  ```
  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
  ```
  Wenn die Spoke-Router einen Lastenausgleich pro Paket durchführen, können Pakete in der falschen Reihenfolge empfangen werden.

Um ein asymmetrisches Routing oder einen Lastenausgleich pro Paket über die Verbindungen zu den beiden Hubs zu vermeiden, müssen Sie das Routing-Protokoll so konfigurieren, dass es einen Spoke-to-Hub-Pfad in beide Richtungen bevorzugt. Wenn Hub1 der primäre und Hub2 der Backup-Switch sein soll, können Sie die Verzögerung für die Hub-Tunnel-Schnittstellen auf andere einstellen.

Hub 1:

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

Hub 2:

```
interface tunnel0
...
```

```
delay 1050
...
```

Hinweis: In diesem Beispiel wurde der Verzögerung bei der Tunnelschnittstelle auf Hub2 50 hinzugefügt, da diese kleiner ist als die Verzögerung bei der Ethernet1-Schnittstelle zwischen den beiden Hubs (100). Auf diese Weise leitet Hub2 Pakete weiterhin direkt an die Spoke-Router weiter, kündigt jedoch eine weniger wünschenswerte Route als Hub1 an Router hinter Hub1 und Hub2 an. Wenn die Verzögerung um mehr als 100 erhöht wurde, leitet Hub2 Pakete für die Spoke-Router über Hub1 über die Ethernet1-Schnittstelle weiter, obwohl die Router hinter Hub1 und Hub2 Hub-1 für das Senden von Paketen an die Spoke-Router immer noch richtig bevorzugen.

Jetzt sehen die Routen wie folgt aus:

Hub 1:

```
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
```

Hub 2:

```
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
```

Die Kosten für die Netzwerkrouten hinter den Spoke-Routern sind bei den beiden Hub-Routern unterschiedlich. In diesem Fall wird Hub1 für die Weiterleitung von Datenverkehr an die Spoke-Router bevorzugt, wie auf R2 zu sehen ist. Dies übernimmt die im ersten Aufzählungspunkt oben beschriebenen Probleme.

Das im zweiten Aufzählungspunkt oben beschriebene Problem ist noch vorhanden, aber da Sie zwei p-pGRE-Tunnelschnittstellen haben, können Sie die **Verzögerung** einstellen.. an den Tunnelschnittstellen separat die EIGRP-Metrik für die von Hub1 und Hub2 übernommenen Routen ändern.

Spoke 1:

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

Spoke2:

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

Jetzt sehen die Routen wie folgt aus:

Spoke 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
```

Spoke2:

```
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
```

Die obige Routing-Konfiguration schützt vor asymmetrischem Routing und ermöglicht gleichzeitig ein Failover auf Hub2, wenn Hub1 ausfällt. Dies bedeutet, dass bei beiden Hubs nur Hub1 verwendet wird.

Wenn Sie beide Hubs verwenden möchten, indem Sie die Stationen über die Hubs hinweg ausbalancieren, mit Failover-Schutz und ohne asymmetrisches Routing, dann ist die Routing-Konfiguration komplexer, aber Sie können dies bei Verwendung von EIGRP tun. Um dies zu erreichen, legen Sie die **Verzögerung ...** auf den Tunnelschnittstellen der Hub-Router zurück auf die Gleichwertigkeit und verwenden Sie dann den Befehl **offset-list <acl> out <offset> <interface>** auf den Spoke-Routern, um die EIGRP-Metrik für Routen zu erhöhen, die über die GRE-Tunnelschnittstellen an den Backup-Hub gemeldet wurden. Die ungleiche **Verzögerung ...** zwischen der Tunnel0-Schnittstelle und der Tunnel1-Schnittstelle am Spoke-Router wird weiterhin verwendet, sodass der Spoke-Router den primären Hub-Router bevorzugt. Die Änderungen an den Spoke-Routern sind wie folgt:

Spoke1-Router

```
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
!
```


```
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
!
```

Hinweis: Der Offset-Wert von 12800 (50*256) wurde der EIGRP-Metrik hinzugefügt, da er kleiner als 25600 (100*256) ist. Dieser Wert (25600) wird der EIGRP-Metrik für Routen hinzugefügt, die zwischen den Hub-Routern erfasst wurden. Wenn der Befehl **Offset-Liste** den Befehl 12800 verwendet, leitet der Backup-Hub-Router Pakete direkt an die Spoke-Router weiter, anstatt diese Pakete über das Ethernet an den primären Hub-Router für diese Spokes weiterzuleiten. Die Metrik für die von den Hub-Routern angekündigten Routen ist weiterhin so beschaffen, dass der richtige primäre Hub-Router bevorzugt wird. Beachten Sie, dass die Hälfte der Spokes Hub1 als primären Router und die andere Hälfte Hub2 als primären Router verwendet.

Hinweis: Wenn der Offset-Wert um mehr als 25600 (100*256) erhöht wurde, würden die Hubs Pakete für die Hälfte der Spoke-Router über die Ethernet1-Schnittstelle über den anderen Hub weiterleiten, obwohl die Router hinter den Hubs weiterhin den richtigen Hub für das Senden von Paketen an die Spoke-Router vorziehen würden.

Hinweis: Der Befehl "**distribute-list 1 out**" wurde ebenfalls hinzugefügt, da Routen, die von einem Hub-Router über eine Tunnelschnittstelle in einem Spoke empfangen wurden, über den anderen Hub angekündigt werden können. Die **Verteilerliste ...** stellt sicher, dass der Spoke-Router nur seine eigenen Routen ankündigen kann.

Hinweis: Wenn Sie die Routing-Meldungen auf den Hub-Routern und nicht auf den Spoke-Routern steuern möchten, können Sie die **Offset-Liste <acl1> in <value> <interface>** und die **distribute-list <acl2> in** Befehlen auf den Hub-Routern statt auf den Spokes konfigurieren. Die Zugriffsliste <acl2> listet die Routen hinter allen Spokes auf, und die Zugriffsliste <acl1> listet nur die Routen hinter den Spokes auf, wobei ein anderer Hub-Router der primäre Hub sein soll.

Bei diesen Änderungen sehen die Routen wie folgt aus:

Hub 1:

```
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
```

Hub 2:

```
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
```

Spoke 1:

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

Spoke2:

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

Schlussfolgerung

Die DMVPN-Lösung bietet folgende Funktionen zur besseren Skalierung großer und kleiner IPsec-VPN-Netzwerke.

- DMVPN ermöglicht eine bessere Skalierung bei Full Mesh- oder partiellen Mesh-IPsec-VPNs.Sie ist besonders nützlich, wenn Spoke-to-Spoke-Datenverkehr sporadisch ist (z. B. sendet jedes Spoke nicht ständig Daten an jedes andere Spoke). Es ermöglicht Spokes, Daten direkt an andere Spokes zu senden, solange direkte IP-Verbindungen zwischen den Spokes bestehen.
- DMVPN unterstützt IPsec-Knoten mit dynamisch zugewiesenen Adressen (z. B. Kabel, ISDN und DSL).Dies gilt für Hub-and-Spoke- und Mesh-Netzwerke. DMVPN kann erfordern, dass die Hub-to-Spoke-Verbindung permanent aktiv ist.
- DMVPN vereinfacht das Hinzufügen von VPN-Knoten.Beim Hinzufügen eines neuen Spoke-Routers müssen Sie nur den Spoke-Router konfigurieren und mit dem Netzwerk verbinden (Sie müssen jedoch möglicherweise ISAKMP-Autorisierungsinformationen für das neue Spoke auf dem Hub hinzufügen). Der Hub erfasst dynamisch die neuen Spokes, und das dynamische Routing-Protokoll leitet das Routing an den Hub und alle anderen Spokes weiter.
- DMVPN reduziert die für alle Router im VPN erforderliche Konfiguration.Dies gilt auch für VPN-Netzwerke mit GRE+IPsec-Hub-and-Spoke-Funktion.
- DMVPN verwendet GRE und unterstützt daher IP-Multicast- und dynamischen Routing-Datenverkehr im gesamten VPN.Dies bedeutet, dass ein dynamisches Routing-Protokoll verwendet werden kann und redundante "Hubs" vom Protokoll unterstützt werden können. Multicast-Anwendungen werden ebenfalls unterstützt.
- DMVPN unterstützt Split-Tunneling an den Stationen.

Zugehörige Informationen

- Dynamic Multipoint VPN (DMVPN)
- IPSec-Support-Seite
- Technischer Support - Cisco Systems