Cisco GRE VPN

Tunneling Overview

Tunnels provide a way to transport protocols that the underlying network does not support. There are several reasons why this may be:

- The network infrastructure doesn't support the protocol being used
- The network infrastructure cannot route the packets due to lack of routing information or addressing types (public addressing vs. private addressing)
- The network infrastructure doesn't support the traffic type (multicast or broadcast)

The most common use case for tunnels is to connect remote, geographically separated sites over an existing network, most notably routing over a public infrastructure (such as the Internet). When used in this manner, tunnels create VPN overlay networks between remote sites. Packets destined to remote private networks are encapsulated within a new IP header that is used to traverse the public internet.

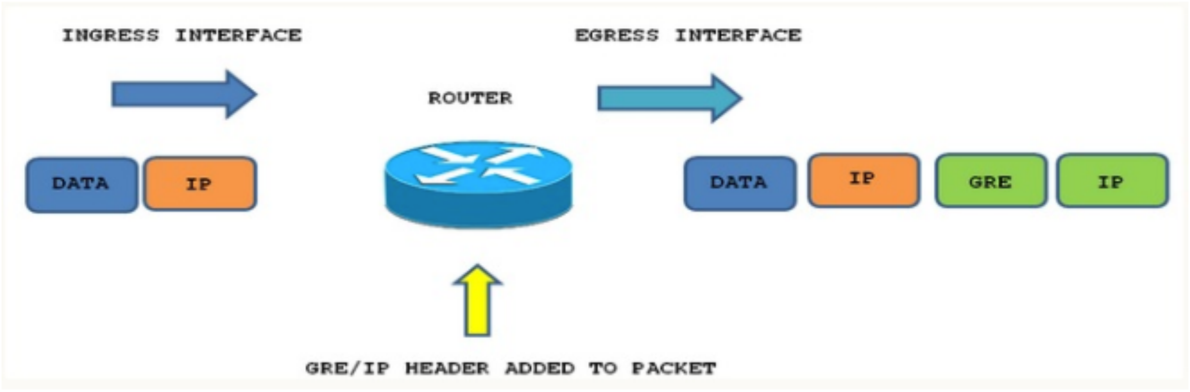
GRE Tunnels

GRE tunnels provide an interface the device can use to forward data. The "data" in this sense is the **passenger protocol** itself, such as IPv6 or IPv4. These tunnels are comprised of three main components:

1. Delivery Header (Transport Protocol)
2. GRE Header (Carrier Protocol)
3. Payload Packet (Passenger Protocol)

GRE can be used with many different combinations of passenger and transport protocols. However, IPv4 and IPv6 are the most common transport protocols for GRE. For example:

- GRE can use IPv4 as the transport protocol to tunnel an IPv4 packet across the underlying network infrastructure.
- GRE can use IPv4 as the transport protocol to tunnel an IPv6 packet across the underlying network infrastructure.
- GRE can use IPv6 as the transport protocol to tunnel an IPv4 packet across the underlying network infrastructure.
- GRE can use IPv6 as the transport protocol to tunnel an IPv6 packet across the underlying network infrastructure.



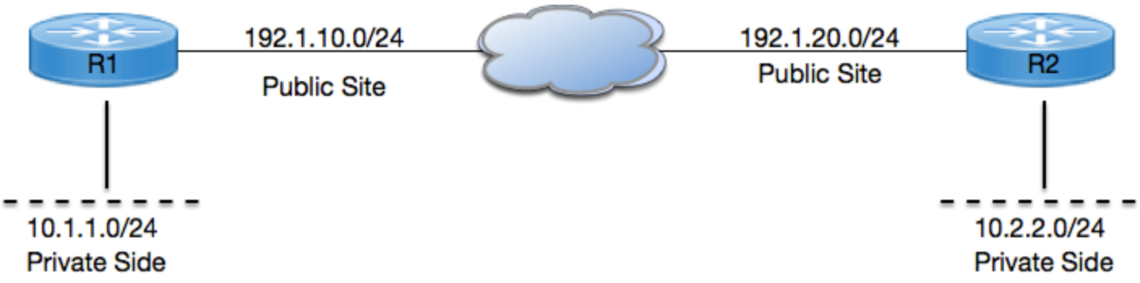
Why Use GRE Tunnels?

GRE's support for multiple protocols and packet types makes it ideal for solving many of the problems faced when trying to form VPNs across the Internet. The most obvious issue is that private addressing used in the enterprise cannot be routed across the public Internet. GRE solves this by encapsulating the IP header with private addressing using an outer IP header that uses public addressing.

GRE can be used to solve both of these problems:

1. GRE supports multicast traffic allowing hello messages generated by an IGP to be transported through the GRE tunnel across the underlying infrastructure as a unicast packet. IPsec can then be used to encrypt all traffic flowing through the GRE tunnel.
2. GRE configuration creates a logical direct connection between two sites over the underlying infrastructure. This means the control plane of the IGP believes it is directly connected to the neighbour with which it is exchanging hellos and therefore can form the adjacency.

Configuration Example



Task-1: Configure GRE VPN between R1 & R2 for making communication between both private sides.

Task-2: Implement GRE Over IPSEC Setup Between R1 and R2 for securing communication between both private side using following parameters:

ISAKMP Parameters:
Encryption: 3DES
Authentication Pre-Share
DH-Group: 2
Hash: MD5
Key: Cisco123

IPSEC Parameters:
Encryption: AES
Hash: SHA

Initial Configuration:

R1:

```
interface FastEthernet0/0
 ip address 192.1.10.1 255.255.255.0
 duplex auto
 speed auto
end
```

```
interface Loopback0
 ip address 10.1.1.1 255.255.255.0
end
```

```
ip route 0.0.0.0 0.0.0.0 192.1.10.2
```

R2:

```
interface FastEthernet0/0
 ip address 192.1.20.3 255.255.255.0
 duplex auto
 speed auto
end
```

```
interface Loopback0
 ip address 10.2.2.2 255.255.255.0
end
```

```
ip route 0.0.0.0 0.0.0.0 192.1.20.2
```

ISP:

```
interface FastEthernet0/0
 ip address 192.1.10.2 255.255.255.0
 duplex auto
 speed auto
end
```

```
interface FastEthernet0/1
 ip address 192.1.20.2 255.255.255.0
 duplex auto
 speed auto
end
```

Task-1: Configuring GRE Tunnel between R1 & R2.

Step-1: Configuring Tunnel interface

Solution:

R1:

```
interface Tunnel1
 ip address 192.168.1.1 255.255.255.0
 tunnel source 192.1.10.1
 tunnel destination 192.1.20.3
end
```

R2:

```
interface Tunnel1
 ip address 192.168.1.2 255.255.255.0
 tunnel source 192.1.20.3
 tunnel destination 192.1.10.1
end
```

Note: Tunnel Source & Destination addresses should be reachable from both ends via some other link (Other than Tunnel interface) like: Wan Links.

Step-2: Configuring Routing between both R1 & R2, for advertising Private Networks and Tunnel Networks.

R1:

```
router eigrp 1
 network 10.0.0.0
 network 192.168.1.0
 no auto-summary
```

R2:

```
router eigrp 1
 network 10.0.0.0
 network 192.168.1.0
 no auto-summary
```

Verification:

```
R1#ping 192.168.1.2

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.1.2, timeout is 2
seconds:
!!!!!
```

```
Success rate is 100 percent (5/5), round-trip min/avg/max =
112/127/144 ms
R2#ping 192.168.1.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.1.1, timeout is 2
seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max =
28/35/52 ms
```

```
R1#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-
2
       ia - IS-IS inter area, * - candidate default, U - per-user static
route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is 192.1.10.2 to network 0.0.0.0

C    192.1.10.0/24 is directly connected, FastEthernet0/0
     10.0.0.0/24 is subnetted, 2 subnets
D       10.2.2.0 [90/297372416] via 192.168.1.2, 00:21:24, Tunnel1
C       10.1.1.0 is directly connected, Loopback0
C    192.168.1.0/24 is directly connected, Tunnel1
S*   0.0.0.0/0 [1/0] via 192.1.10.2
```

```
R2#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-
2
       ia - IS-IS inter area, * - candidate default, U - per-user static
route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is 192.1.20.2 to network 0.0.0.0

C    192.1.20.0/24 is directly connected, FastEthernet0/0
     10.0.0.0/24 is subnetted, 2 subnets
C       10.2.2.0 is directly connected, Loopback0
D       10.1.1.0 [90/297372416] via 192.168.1.1, 00:21:18, Tunnel1
C    192.168.1.0/24 is directly connected, Tunnel1
S*   0.0.0.0/0 [1/0] via 192.1.20.2
```

```
R1#ping 10.2.2.2

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.2.2.2, timeout is 2
seconds:
!!!!!
```

```
Success rate is 100 percent (5/5), round-trip min/avg/max =
108/112/116 ms
```

```
R2#ping 10.1.1.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.1.1, timeout is 2
seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max =
36/41/48 ms
```

Task-2: Implement GRE Over IPSEC Setup Between R1 and R2 for securing communication between both private side using following parameters:

Solution:

R1:

Step-1: Configure Phase-1 ISAKMP Policy:

```
crypto isakmp policy 1
 encr 3des
 hash md5
 authentication pre-share
 group 2
crypto isakmp key cisco123 address 192.1.20.3
```

Step-2: Configure Phase – 2 IPSEC Transform-Set

```
crypto ipsec transform-set TSET esp-3des esp-sha-hmac
```

Step-3: Configure IPSEC Profile

```
crypto ipsec profile IPSEC-PROF
 set transform-set TSET
```

Step-4: Apply IPSEC Profile under Tunnel interface

```
interface Tunnel1
 tunnel protection ipsec profile IPSEC-PROF
end
```

R2:

Step-1: Configure Phase-1 ISAKMP Policy:

```
crypto isakmp policy 1
 encr 3des
 hash md5
 authentication pre-share
 group 2
crypto isakmp key cisco123 address 192.1.10.1
```

Step-2: Configure Phase – 2 IPSEC Transform-Set

```
crypto ipsec transform-set TSET esp-3des esp-sha-hmac
```

Step-3: Configure IPSEC Profile

```
crypto ipsec profile IPSEC-PROF
 set transform-set TSET
```

Step-4: Apply IPSEC Profile under Tunnel interface

```
interface Tunnel1
 tunnel protection ipsec profile IPSEC-PROF
end
```

Verification:

```
R1#show crypto isakmp sa
IPv4 Crypto ISAKMP SA
dst             src             state          conn-id slot status
192.1.20.3      192.1.10.1      QM_IDLE           1001    0 ACTIVE
```

```
R2#show crypto isakmp sa
IPv4 Crypto ISAKMP SA
dst             src             state          conn-id slot status
192.1.20.3      192.1.10.1      QM_IDLE           1001    0 ACTIVE
```

```
R1#show crypto ipsec sa

interface: Tunnel1
    Crypto map tag: Tunnel1-head-0, local addr 192.1.10.1

   protected vrf: (none)
   local  ident (addr/mask/prot/port): (192.1.10.1/255.255.255.255/47/0)
   remote ident (addr/mask/prot/port): (192.1.20.3/255.255.255.255/47/0)
   current_peer 192.1.20.3 port 500
     PERMIT, flags={origin_is_acl,}
    #pkts encaps: 613, #pkts encrypt: 613, #pkts digest: 613
    #pkts decaps: 607, #pkts decrypt: 607, #pkts verify: 607
    #pkts compressed: 0, #pkts decompressed: 0
    #pkts not compressed: 0, #pkts compr. failed: 0
    #pkts not decompressed: 0, #pkts decompress failed: 0
    #send errors 2, #recv errors 0

     local crypto endpt.: 192.1.10.1, remote crypto endpt.: 192.1.20.3
     path mtu 1500, ip mtu 1500, ip mtu idb FastEthernet0/0
     current outbound spi: 0xC14FAFC3(3243225027)

     inbound esp sas:
      spi: 0xA739CCF6(2805583094)
        transform: esp-3des esp-sha-hmac ,
        in use settings ={Tunnel, }
        conn id: 1, flow_id: SW:1, crypto map: Tunnel1-head-0
        sa timing: remaining key lifetime (k/sec): (4567902/873)
        IV size: 8 bytes
        replay detection support: Y
        Status: ACTIVE

     inbound ah sas:

     inbound pcp sas:

     outbound esp sas:
      spi: 0xC14FAFC3(3243225027)
        transform: esp-3des esp-sha-hmac ,
        in use settings ={Tunnel, }
        conn id: 2, flow_id: SW:2, crypto map: Tunnel1-head-0
        sa timing: remaining key lifetime (k/sec): (4567901/873)
        IV size: 8 bytes
        replay detection support: Y
        Status: ACTIVE
```

```
R2#show crypto ipsec sa

interface: Tunnel1
    Crypto map tag: Tunnel1-head-0, local addr 192.1.20.3

   protected vrf: (none)
   local  ident (addr/mask/prot/port): (192.1.20.3/255.255.255.255/47/0)
   remote ident (addr/mask/prot/port): (192.1.10.1/255.255.255.255/47/0)
   current_peer 192.1.10.1 port 500
     PERMIT, flags={origin_is_acl,}
    #pkts encaps: 617, #pkts encrypt: 617, #pkts digest: 617
    #pkts decaps: 623, #pkts decrypt: 623, #pkts verify: 623
    #pkts compressed: 0, #pkts decompressed: 0
    #pkts not compressed: 0, #pkts compr. failed: 0
    #pkts not decompressed: 0, #pkts decompress failed: 0
    #send errors 0, #recv errors 0

     local crypto endpt.: 192.1.20.3, remote crypto endpt.: 192.1.10.1
     path mtu 1500, ip mtu 1500, ip mtu idb FastEthernet0/0
     current outbound spi: 0xA739CCF6(2805583094)

     inbound esp sas:
      spi: 0xC14FAFC3(3243225027)
        transform: esp-3des esp-sha-hmac ,
        in use settings ={Tunnel, }
        conn id: 1, flow_id: SW:1, crypto map: Tunnel1-head-0
        sa timing: remaining key lifetime (k/sec): (4511307/867)
        IV size: 8 bytes
        replay detection support: Y
        Status: ACTIVE

     inbound ah sas:

     inbound pcp sas:

     outbound esp sas:
      spi: 0xA739CCF6(2805583094)
```

```
        transform: esp-3des esp-sha-hmac ,
        in use settings ={Tunnel, }
        conn id: 2, flow_id: SW:2, crypto map: Tunnel1-head-0
        sa timing: remaining key lifetime (k/sec): (4511308/867)
        IV size: 8 bytes
        replay detection support: Y
        Status: ACTIVE
```

```
R1#show interfaces tunnel 1
Tunnel1 is up, line protocol is up
  Hardware is Tunnel
  Internet address is 192.168.1.1/24
  MTU 1514 bytes, BW 9 Kbit/sec, DLY 500000 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation TUNNEL, loopback not set
  Keepalive not set
  Tunnel source 192.1.10.1, destination 192.1.20.3
  Tunnel protocol/transport GRE/IP
    Key disabled, sequencing disabled
    Checksumming of packets disabled
  Tunnel TTL 255
  Fast tunneling enabled
  Tunnel transmit bandwidth 8000 (kbps)
  Tunnel receive bandwidth 8000 (kbps)
  Tunnel protection via IPSec (profile "IPSEC-PROF")
  Last input 00:00:00, output never, output hang never
  Last clearing of "show interface" counters never
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 3
  Queueing strategy: fifo
  Output queue: 0/0 (size/max)
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
     627 packets input, 53444 bytes, 0 no buffer
     Received 0 broadcasts, 0 runts, 0 giants, 0 throttles
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
     633 packets output, 53968 bytes, 0 underruns
     0 output errors, 0 collisions, 0 interface resets
     0 unknown protocol drops
     0 output buffer failures, 0 output buffers swapped out
```

```
R2#show interfaces tunnel 1
Tunnel1 is up, line protocol is up
  Hardware is Tunnel
  Internet address is 192.168.1.2/24
  MTU 1514 bytes, BW 9 Kbit/sec, DLY 500000 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation TUNNEL, loopback not set
  Keepalive not set
  Tunnel source 192.1.20.3, destination 192.1.10.1
  Tunnel protocol/transport GRE/IP
    Key disabled, sequencing disabled
    Checksumming of packets disabled
  Tunnel TTL 255
  Fast tunneling enabled
  Tunnel transmit bandwidth 8000 (kbps)
  Tunnel receive bandwidth 8000 (kbps)
  Tunnel protection via IPSec (profile "IPSEC-PROF")
  Last input 00:00:03, output never, output hang never
  Last clearing of "show interface" counters never
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 1
  Queueing strategy: fifo
  Output queue: 0/0 (size/max)
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
     639 packets input, 54472 bytes, 0 no buffer
     Received 0 broadcasts, 0 runts, 0 giants, 0 throttles
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
     633 packets output, 53948 bytes, 0 underruns
     0 output errors, 0 collisions, 0 interface resets
     0 unknown protocol drops
     0 output buffer failures, 0 output buffers swapped out
```

```
R1#ping 192.168.1.2

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.1.2, timeout is 2
seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max =
32/50/76 ms
```

```
R2#ping 192.168.1.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.1.1, timeout is 2
seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max =
28/33/40 ms
R2#
```

```
R1#ping 10.2.2.2

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.2.2.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max =
20/33/60 ms
```

Cisco DMVPN

Dynamic Multipoint VPN (DMVPN) is a Cisco VPN solution used when high scalability and minimal configuration complexity is required in connecting branch offices to a central HQ Hub site.

DMVPN is one of the most scalable and most efficient VPN types supported by Cisco. It is used almost exclusively with Hub-and-Spoke topologies where you want to have direct Spoke-to-Spoke VPN tunnels in addition to the Spoke-to-Hub tunnels. This means that Spoke sites can communicate between them directly without having to go through the Hub. DMVPN is supported only on Cisco Routers.

If you want to design a VPN solution to connect numerous sites between them (I would say more than 10 sites), then DMVPN using Cisco routers is an ideal choice. Although the most common topology is Hub-and-spoke setup, DMVPN supports full mesh connectivity since all sites can communicate between them without having to configure static VPN tunnels between each other.

Some characteristics of DMVPN are the following:

- • The HUB central router acts as the DMVPN server and the Spoke routers (in branch offices) act as the DMVPN clients.

- • The HUB router must have static public IP address on its WAN interface.

- • The spoke branch routers can have either static or dynamic public IP on the WAN.

- • Each branch site (Spoke) has a permanent IPSEC Tunnel with the Central site (Hub).

- • The Spoke-to-Spoke tunnels are established on demand whenever there is traffic between the Spoke sites. Thereafter, packets are able to bypass the Hub site and use the spoke-to-spoke tunnel directly.

- • All tunnels are using Multipoint GRE with IPSEC Protection.

- • NHRP (Next Hop Resolution Protocol) is used to map the private IPs of Tunnel Interfaces with their corresponding WAN Public IPs.

- • The above NHRP mappings will be kept on the NHRP Server router (HUB). Each Spoke communicates with the NHRP Server (Hub) and registers its public IP address and its private Tunnel Interface IP to the Hub router. Thus, the Hub router will store all mappings for "**Tunnel Interface IP / Public WAN IP**" of all the Spoke sites.

- • When a spoke needs to send a packet to a destination (private) subnet on another spoke, it queries the NHRP server in order to learn the public (outside WAN) address of the destination (target) spoke.

- • For better scalability, it is recommended to run a dynamic routing protocols (such as EIGRP) between all the routers.

<u>DMVPN uses the following group of networking technologies</u>

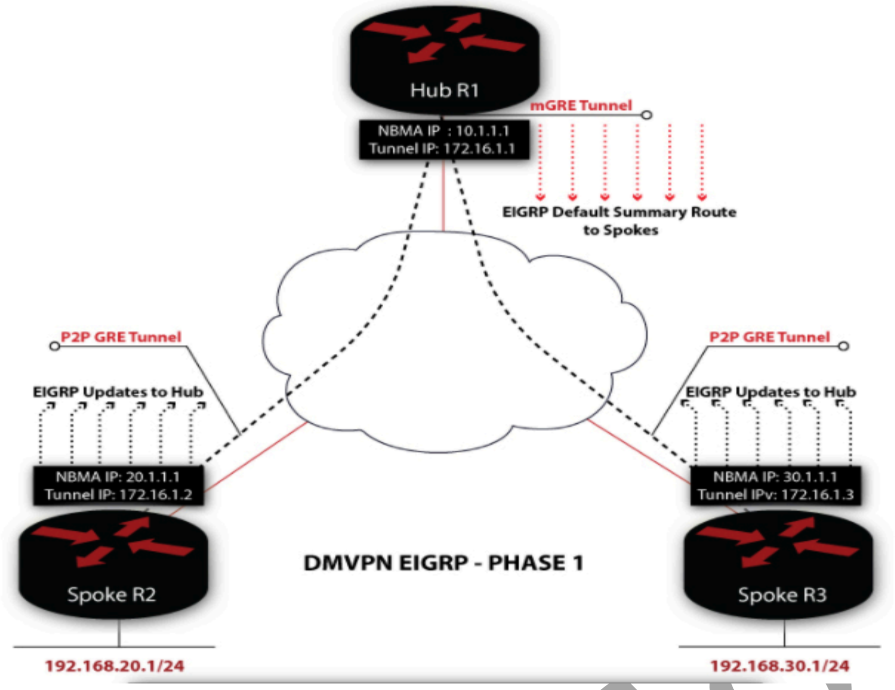
- • Multipoint GRE
- • IPSEC
- • Next-Hop Resolution Protocol – NHRP
- • Static or dynamic routing

DMVPN have three phases:

Phase 1

DMVPN phase 1 only provides hub-and-spoke tunnel deployment. This means GRE tunnels are only built between the hub and the spokes. Traffic destined to networks behind spokes is forced to first traverse the hub.

The topology below shows two spokes connected to the hub router. The hub is configured with an mGRE tunnel and the spokes with a P2P GRE tunnel.



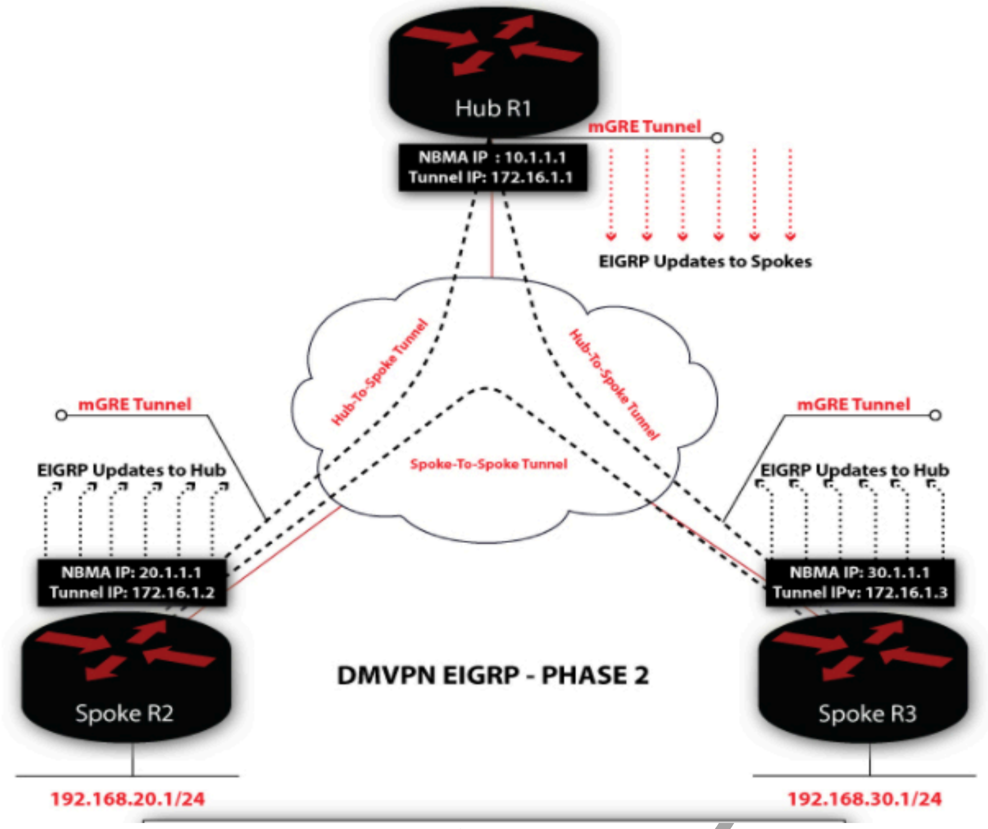
There are two critical configurations that make this a Phase 1 implementation:

1. Configuring the spoke's tunnel interface as P2P GRE tunnel (In all phases, the hub is always configured with an mGRE tunnel)
2. The next hop on the spokes always point towards the hub

Phase 2

In Phase 1, traffic between the spokes would always hit the hub. This was a shortcoming of DMVPN as, in a larger deployment, the hub would always have to be burdened with encapsulate/decapsulate overhead for the spoke-to-spoke traffic. In addition to increased routing overhead on the hub, spoke-to-spoke traffic would take a suboptimal path by detouring to the hub and then reaching the remote spoke. Phase 2 improved on Phase 1 by allowing spokes to build a spoke-to-spoke tunnel on demand with these restrictions:

- Spokes must use multipoint GRE tunnels
- The spokes must receive specific routes for all remote spoke subnets
- The next hop of the entry in the routing table must list the remote spoke as the next hop



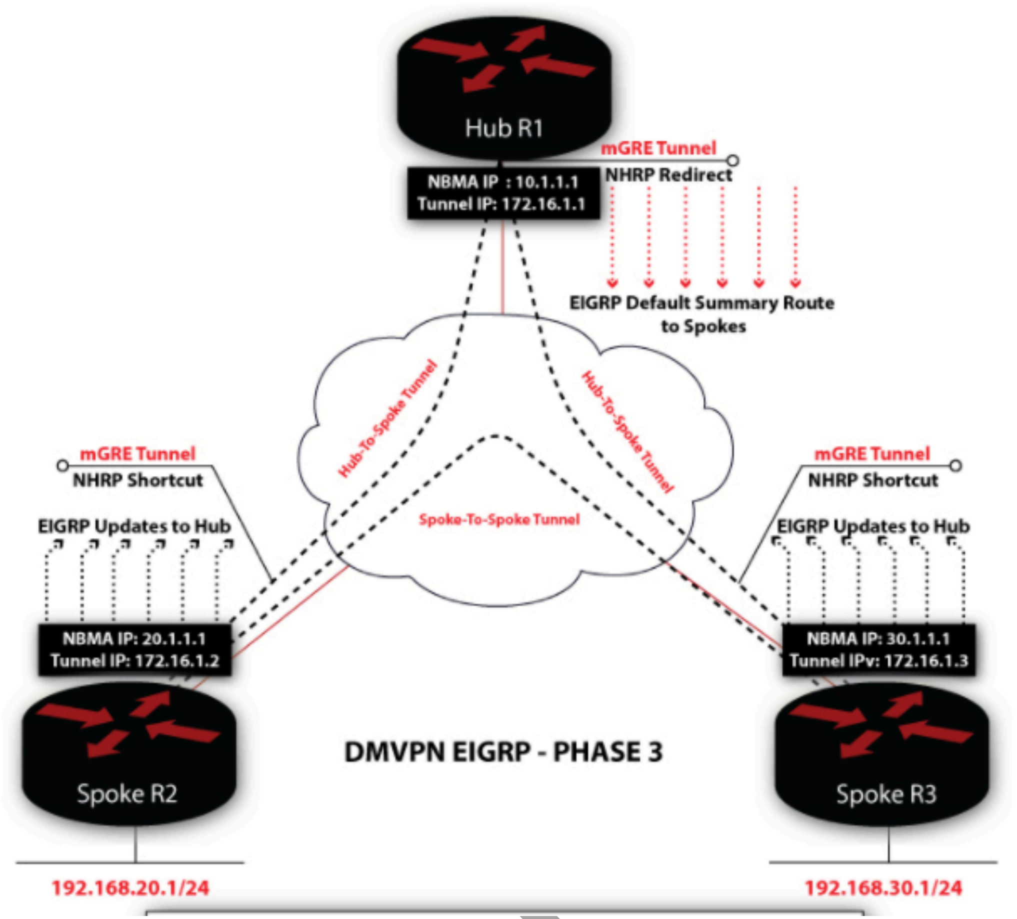
First, it must be ensured the spokes utilize multipoint GRE tunnels. Configuring mGRE on the Spokes allows multiple GRE tunnels to be formed using a single tunnel interface. This is achieved by removing the static **tunnel destination** command and replacing it with the **tunnel mode gre multipoint** command.

Second, the spokes must receive specific routes for all remote spoke subnets. For EIGRP, this is accomplished by disabling split horizon on the tunnel interface. The split-horizon algorithm is, “Do not advertise a route out an interface if the router uses that interface to reach that network.”

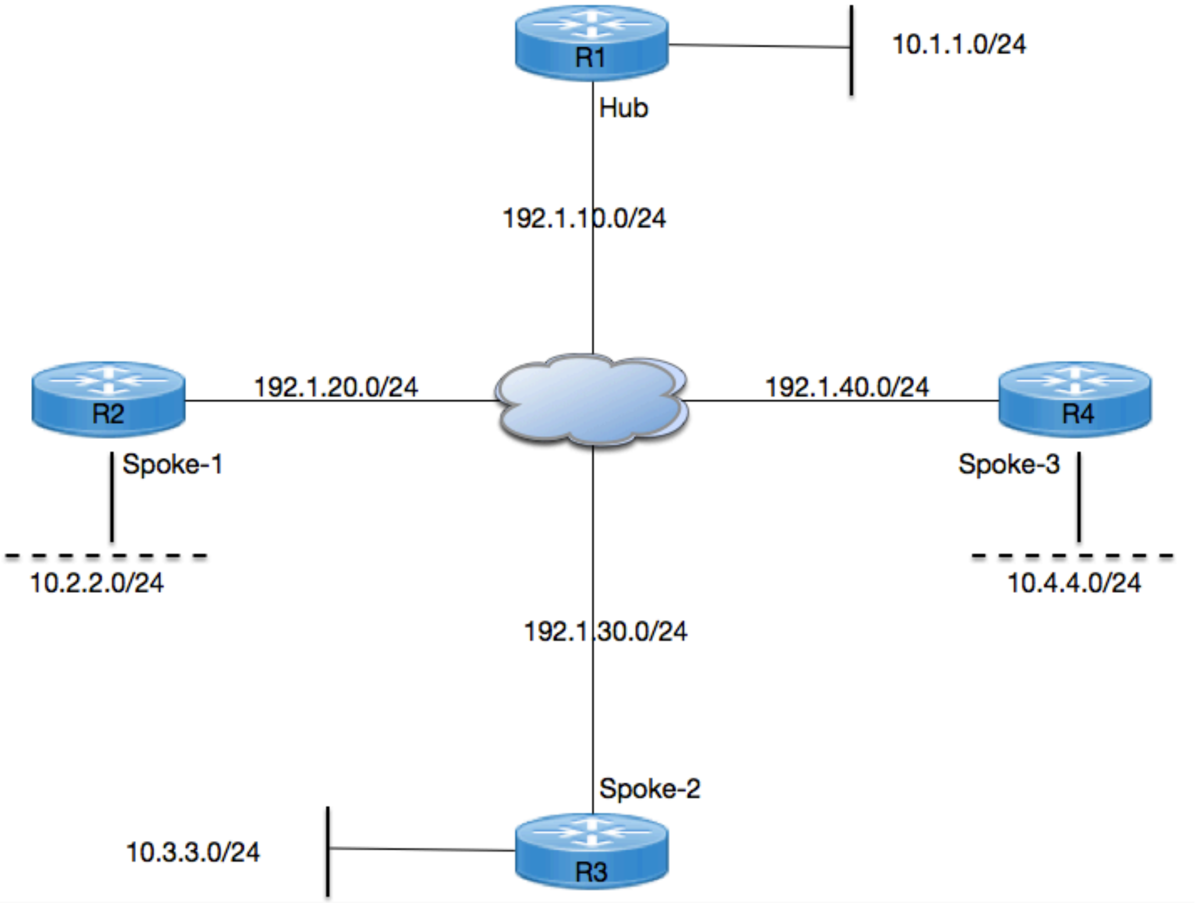
Phase 3

Though DMVPN Phase 2 deployment provided direct spoke-to-spoke tunnels, one of the limitations is maintaining full routing tables on the spokes. Each route for remote spoke networks needs to be a specific route with the next hop pointing to the remote spoke’s tunnel address. This prevents the hub from being able to send down a summarized route to the spokes for a more concise routing table.

Phase 3 overcomes this restriction using NHRP traffic indication messages from the hub to signal to the spokes that a better path exists to reach the target network. This functionality is enabled by configuring **ip nhrp redirect** on the hub and **ip nhrp shortcut** on the spokes. The redirect command tells the hub to send the NHRP traffic indication message while the shortcut command tells the spokes to accept the redirect and install the shortcut route.


Hub R1
mGRE Tunnel
NHRP Redirect
NBMA IP : 10.1.1.1
Tunnel IP: 172.16.1.1
EIGRP Default Summary Route
to Spokes
Hub-To-Spoke Tunnel
Hub-To-Spoke Tunnel
Spoke-To-Spoke Tunnel
mGRE Tunnel
NHRP Shortcut
EIGRP Updates to Hub
NBMA IP: 20.1.1.1
Tunnel IP: 172.16.1.2
mGRE Tunnel
NHRP Shortcut
EIGRP Updates to Hub
NBMA IP: 30.1.1.1
Tunnel IPv: 172.16.1.3
DMVPN EIGRP - PHASE 3
Spoke R2
Spoke R3
192.168.20.1/24
192.168.30.1/24

Configuration Example



Task-1: Configure MGRE Tunnels between R1, R2, R3 & R4 and assign ip address to tunnel interfaces from 192.168.1.0/24 Subnet.

Task-2: Configure NHRP between tunnel interfaces of R1, R2, R3 & R4 in which configure R1 as Hub and R2, R3 & R4 as Spoke 1, 2 & 3 respectively.

Task-3: Configure IPSEC between Hub and Spokes with below parameters, and configure DMVPN Phase 1, 2 & 3.

ISAKMP Parameters:
Encryption: 3DES
Hash: MD5
DH Group: 2
Authentication: Pre-Share
Key: Cisco123

IPSEC Parameters:
Encryption: AES
Hash: SHA

Initial Configuration:

R1:

```
interface FastEthernet0/0
 ip address 192.1.10.1 255.255.255.0
 duplex auto
 speed auto
end
```

```
interface Loopback0
 ip address 10.1.1.1 255.255.255.0
end
```

```
ip route 0.0.0.0 0.0.0.0 192.1.10.5
```

R2:

```
interface FastEthernet0/0
 ip address 192.1.20.2 255.255.255.0
 duplex auto
 speed auto
end
```

```
interface Loopback0
 ip address 10.2.2.2 255.255.255.0
end
```

```
ip route 0.0.0.0 0.0.0.0 192.1.20.5
```

R3:

```
interface FastEthernet1/0
 ip address 192.1.30.3 255.255.255.0
 duplex auto
 speed auto
end
```

```
interface Loopback0
 ip address 10.3.3.3 255.255.255.0
end
```

```
ip route 0.0.0.0 0.0.0.0 192.1.30.5
```

R4:

```
interface FastEthernet2/0
 ip address 192.1.40.4 255.255.255.0
 duplex auto
 speed auto
end
```

```
interface Loopback0
```

```
 ip address 10.4.4.4 255.255.255.0
end
ip route 0.0.0.0 0.0.0.0 192.1.40.5
```

ISP:

```
interface FastEthernet0/0
 ip address 192.1.10.5 255.255.255.0
 duplex auto
 speed auto
end
```

```
interface FastEthernet0/1
 ip address 192.1.20.5 255.255.255.0
 duplex auto
 speed auto
end
```

```
interface FastEthernet1/0
 ip address 192.1.30.5 255.255.255.0
 duplex auto
 speed auto
end
```

```
interface FastEthernet2/0
 ip address 192.1.40.5 255.255.255.0
 duplex auto
 speed auto
end
```

Task-1: Configure MGRE Tunnels between R1, R2, R3 & R4 and assign ip address to tunnel interfaces from 192.168.1.0/24 Subnet.

Solution:

R1:

```
interface Tunnel1
 ip address 192.168.1.1 255.255.255.0
 no ip redirects
 no ip split-horizon eigrp 1
 tunnel source 192.1.10.1
 tunnel mode gre multipoint
```

R2:

```
interface Tunnel1
 ip address 192.168.1.2 255.255.255.0
 no ip redirects
 tunnel source 192.1.20.2
 tunnel mode gre multipoint
end
```

R3:

```
interface Tunnel1
 ip address 192.168.1.3 255.255.255.0
 no ip redirects
 tunnel source 192.1.30.3
 tunnel mode gre multipoint
end
```

R4:

```
interface Tunnel1
 ip address 192.168.1.4 255.255.255.0
 no ip redirects
 tunnel source 192.1.40.4
 tunnel mode gre multipoint
```

Task-2: Configure NHRP between tunnel interfaces of R1, R2, R3 & R4 in which configure R1 as Hub and R2, R3 & R4 as Spoke 1, 2 & 3 respectively.

Solution:

R1 (Hub):

```
interface Tunnel1
 ip nhrp authentication cisco
 ip nhrp map multicast dynamic
 ip nhrp network-id 123
 ip nhrp redirect
end
```

R2 (Spoke-1):

```
interface Tunnel1
 ip nhrp authentication cisco
 ip nhrp map 192.168.1.1 192.1.10.1
 ip nhrp map multicast 192.1.10.1
 ip nhrp network-id 123
 ip nhrp nhs 192.168.1.1
 ip nhrp shortcut
end
```

R3 (Spoke-2):

```
interface Tunnel1
 ip nhrp authentication cisco
 ip nhrp map 192.168.1.1 192.1.10.1
 ip nhrp map multicast 192.1.10.1
 ip nhrp network-id 123
 ip nhrp nhs 192.168.1.1
 ip nhrp shortcut
end
```

R4 (Spoke-3):

```
interface Tunnel1
 ip nhrp authentication cisco
 ip nhrp map 192.168.1.1 192.1.10.1
 ip nhrp map multicast 192.1.10.1
 ip nhrp network-id 123
 ip nhrp nhs 192.168.1.1
 ip nhrp shortcut
end
```

Task-3: Configure IPSEC between Hub and Spokes with below parameters, and configure DMVPN Phase 1, 2 & 3.

Solution:

R1:

Step-1: Configuring Phase 1 ISAKMP:

```
crypto isakmp policy 1
 encr 3des
 hash md5
 authentication pre-share
 group 2
crypto isakmp key cisco123 address 0.0.0.0 0.0.0.0
```

Step-2: Configure Phase 2 IPSEC:

```
crypto ipsec transform-set TSET esp-aes esp-sha-hmac
```

Step-3: Configure IPSEC Profile:

```
crypto ipsec profile IPSEC-PROF
 set transform-set TSET
```

Step-4: Apply IPSEC Profile under tunnel interface:

```
interface Tunnel1
tunnel protection ipsec profile IPSEC-PROF
end
```

Step-5: Configure EIGRP on tunnel interface and advertise Lan Interface in that:

```
router eigrp 1
 network 10.0.0.0
 network 192.168.1.0
 no auto-summary
```

R2:

Step-1: Configuring Phase 1 ISAKMP:

```
crypto isakmp policy 1
 encr 3des
 hash md5
 authentication pre-share
 group 2
crypto isakmp key cisco123 address 0.0.0.0 0.0.0.0
```

Step-2: Configure Phase 2 IPSEC:

```
crypto ipsec transform-set TSET esp-aes esp-sha-hmac
```

Step-3: Configure IPSEC Profile:

```
crypto ipsec profile IPSEC-PROF
 set transform-set TSET
```

Step-4: Apply IPSEC Profile under tunnel interface:

```
interface Tunnel1
tunnel protection ipsec profile IPSEC-PROF
end
```

Step-5: Configure EIGRP on tunnel interface and advertise Lan Interface in that:

```
router eigrp 1
 network 10.0.0.0
 network 192.168.1.0
 no auto-summary
```

R3:

Step-1: Configuring Phase 1 ISAKMP:

```
crypto isakmp policy 1
 encr 3des
 hash md5
 authentication pre-share
 group 2
crypto isakmp key cisco123 address 0.0.0.0 0.0.0.0
```

Step-2: Configure Phase 2 IPSEC:

```
crypto ipsec transform-set TSET esp-aes esp-sha-hmac
```

Step-3: Configure IPSEC Profile:

```
crypto ipsec profile IPSEC-PROF
 set transform-set TSET
```

Step-4: Apply IPSEC Profile under tunnel interface:

```
interface Tunnel1
tunnel protection ipsec profile IPSEC-PROF
end
```

Step-5: Configure EIGRP on tunnel interface and advertise Lan Interface in that:

```
router eigrp 1
 network 10.0.0.0
 network 192.168.1.0
 no auto-summary
```

R4:

Step-1: Configuring Phase 1 ISAKMP:

```
crypto isakmp policy 1
 encr 3des
 hash md5
 authentication pre-share
 group 2
crypto isakmp key cisco123 address 0.0.0.0 0.0.0.0
```

Step-2: Configure Phase 2 IPSEC:

```
crypto ipsec transform-set TSET esp-aes esp-sha-hmac
```

Step-3: Configure IPSEC Profile:

```
crypto ipsec profile IPSEC-PROF
 set transform-set TSET
```

Step-4: Apply IPSEC Profile under tunnel interface:

```
interface Tunnel1
tunnel protection ipsec profile IPSEC-PROF
end
```

Step-5: Configure EIGRP on tunnel interface and advertise Lan Interface in that:

```
router eigrp 1
 network 10.0.0.0
 network 192.168.1.0
 no auto-summary
```

DMVPN Phase – 1:

Configuration:

R1:

```
interface Tunnel1
no ip split-horizon eigrp 1
end
```

R2:

```
interface Tunnel1
no ip split-horizon eigrp 1
end
```

R3:

```
interface Tunnel1
no ip split-horizon eigrp 1
end
```

R4:

```
interface Tunnel1
no ip split-horizon eigrp 1
end
```

Verification:

R1 (Hub):

```
R1#show ip nhrp
192.168.1.2/32 via 192.168.1.2, Tunnel1 created 00:29:26, expire
01:30:34
  Type: dynamic, Flags: unique registered
  NBMA address: 192.1.20.2
192.168.1.3/32 via 192.168.1.3, Tunnel1 created 00:29:17, expire
01:30:42
  Type: dynamic, Flags: unique registered
  NBMA address: 192.1.30.3
192.168.1.4/32 via 192.168.1.4, Tunnel1 created 00:29:25, expire
01:30:34
  Type: dynamic, Flags: unique registered
  NBMA address: 192.1.40.4
```

```
R1#show dmvpn
Legend: Attrb --> S - Static, D - Dynamic, I - Incompletea
      N - NATed, L - Local, X - No Socket
      # Ent --> Number of NHRP entries with same NBMA peer

Tunnel1, Type:Hub, NHRP Peers:3,
 # Ent  Peer NBMA Addr Peer Tunnel Add State  UpDn Tm Attrb
 ----- --------------- --------------- ----- -------- -----
```

```
     1      192.1.20.2     192.168.1.2    UP    never D
     1      192.1.30.3     192.168.1.3    UP    never D
     1      192.1.40.4     192.168.1.4    UP    never D
```

```
R1#show interfaces tunnel 1
Tunnel1 is up, line protocol is up
  Hardware is Tunnel
  Internet address is 192.168.1.1/24
  MTU 1514 bytes, BW 9 Kbit/sec, DLY 500000 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation TUNNEL, loopback not set
  Keepalive not set
  Tunnel source 192.1.10.1, destination UNKNOWN
  Tunnel protocol/transport multi-GRE/IP
    Key disabled, sequencing disabled
    Checksumming of packets disabled
  Tunnel TTL 255
  Fast tunneling enabled
  Tunnel transmit bandwidth 8000 (kbps)
  Tunnel receive bandwidth 8000 (kbps)
  Tunnel protection via IPSec (profile "IPSEC-PROF")
  Last input 00:00:03, output 00:00:03, output hang never
  Last clearing of "show interface" counters never
  Input queue: 0/75/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: fifo
  Output queue: 0/0 (size/max)
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
     1137 packets input, 95631 bytes, 0 no buffer
     Received 0 broadcasts, 0 runts, 0 giants, 0 throttles
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
     1211 packets output, 102266 bytes, 0 underruns
     0 output errors, 0 collisions, 0 interface resets
     0 unknown protocol drops
     0 output buffer failures, 0 output buffers swapped out
```

```
R1#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS
level-2
       ia - IS-IS inter area, * - candidate default, U - per-user
static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is 192.1.10.5 to network 0.0.0.0

C    192.1.10.0/24 is directly connected, FastEthernet0/0
     10.0.0.0/24 is subnetted, 4 subnets
D       10.4.4.0 [90/297372416] via 192.168.1.4, 00:31:47, Tunnel1
D       10.3.3.0 [90/297372416] via 192.168.1.3, 00:31:44, Tunnel1
D       10.2.2.0 [90/297372416] via 192.168.1.2, 00:31:49, Tunnel1
C       10.1.1.0 is directly connected, Loopback0
C    192.168.1.0/24 is directly connected, Tunnel1
S*   0.0.0.0/0 [1/0] via 192.1.10.5
```

```
R1#ping 10.2.2.2

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.2.2.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 20/37/64 ms
```

```
R1#ping 10.3.3.3

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.3.3.3, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 32/39/60 ms
R1#
```

```
R1#ping 10.4.4.4

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.4.4.4, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 36/45/60 ms
R1#
```

R2 (Spoke-1):

```
R2#show ip nhrp
192.168.1.1/32 via 192.168.1.1, Tunnel1 created 00:32:01, never expire
  Type: static, Flags: used
  NBMA address: 192.1.10.1
```

```
R2#show dmvpn
Legend: Attrb --> S - Static, D - Dynamic, I - Incompletea
      N - NATed, L - Local, X - No Socket
      # Ent --> Number of NHRP entries with same NBMA peer

Tunnel1, Type:Spoke, NHRP Peers:1,
 # Ent  Peer NBMA Addr Peer Tunnel Add State  UpDn Tm Attrb
 ----- --------------- --------------- ----- -------- -----
     1      192.1.10.1     192.168.1.1    UP 00:32:17 S

```

```
R2#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS
level-2
       ia - IS-IS inter area, * - candidate default, U - per-user
static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is 192.1.20.5 to network 0.0.0.0

C    192.1.20.0/24 is directly connected, FastEthernet0/0
     10.0.0.0/24 is subnetted, 4 subnets
D       10.4.4.0 [90/310172416] via 192.168.1.1, 00:00:05, Tunnel1
```

```
D        10.3.3.0 [90/310172416] via 192.168.1.1, 00:00:05, Tunnel1
C        10.2.2.0 is directly connected, Loopback0
D        10.1.1.0 [90/297372416] via 192.168.1.1, 00:00:11, Tunnel1
C     192.168.1.0/24 is directly connected, Tunnel1
S*    0.0.0.0/0 [1/0] via 192.1.20.5
```

```
R2#ping 10.1.1.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.1.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 32/40/48 ms
```

```
R2#ping 10.3.3.3

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.3.3.3, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 52/93/120
ms
```

```
R2#ping 10.4.4.4

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.4.4.4, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 60/98/148
ms
```

```
R2#traceroute 10.3.3.3

Type escape sequence to abort.
Tracing the route to 10.3.3.3

  1 192.168.1.1 56 msec 40 msec 32 msec
  2 192.168.1.3 72 msec 64 msec 48 msec
```

```
R2#traceroute 10.4.4.4

Type escape sequence to abort.
Tracing the route to 10.4.4.4

  1 192.168.1.1 120 msec 68 msec 40 msec
  2 192.168.1.4 80 msec 124 msec 84 msec
```

Note: You can check other spoke outputs as well.

DMVPN Phase 2:

Configuration:
R1 (Hub):

```
interface Tunnel1
no ip next-hop-self eigrp 1
end
```

Verification:

```
R2#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS
level-2
       ia - IS-IS inter area, * - candidate default, U - per-user
static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is 192.1.20.5 to network 0.0.0.0

C    192.1.20.0/24 is directly connected, FastEthernet0/0
     10.0.0.0/24 is subnetted, 4 subnets
D       10.4.4.0 [90/310172416] via 192.168.1.4, 00:01:10, Tunnel1
D       10.3.3.0 [90/310172416] via 192.168.1.3, 00:01:10, Tunnel1
C       10.2.2.0 is directly connected, Loopback0
D       10.1.1.0 [90/297372416] via 192.168.1.1, 00:01:11, Tunnel1
C    192.168.1.0/24 is directly connected, Tunnel1
S*   0.0.0.0/0 [1/0] via 192.1.20.5
```

```
R2#traceroute 10.3.3.3

Type escape sequence to abort.
Tracing the route to 10.3.3.3

  1 192.168.1.3 152 msec 112 msec 160 msec
```

```
R2#traceroute 10.4.4.4

Type escape sequence to abort.
Tracing the route to 10.4.4.4

  1 192.168.1.4 160 msec 120 msec 116 msec
```

Note: You can check other spoke outputs as well.

DMVPN Phase 3:

Configuration:

R1 (Hub):

```
interface Tunnel1
ip nhrp redirect
end
```

R2 (Spoke-1):

```
interface Tunnel1
ip nhrp shortcut
end
```

R3 (Spoke-2):

```
interface Tunnel1
ip nhrp shortcut
end
```

R4 (Spoke-3):

```
interface Tunnel1
ip nhrp shortcut
end
```

Verification:

- Revert Back IP Next-Hope-Self on R1 (Hub).

```
R1(config)#interface tunnel 1
R1(config-if)#ip next-hop-self eigrp 1
R1(config-if)#end
R1#
```

```
R2#show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS
level-2
       ia - IS-IS inter area, * - candidate default, U - per-user
static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is 192.1.20.5 to network 0.0.0.0

C    192.1.20.0/24 is directly connected, FastEthernet0/0
     10.0.0.0/24 is subnetted, 4 subnets
D       10.4.4.0 [90/310172416] via 192.168.1.1, 00:01:05, Tunnel1
D       10.3.3.0 [90/310172416] via 192.168.1.1, 00:01:04, Tunnel1
C       10.2.2.0 is directly connected, Loopback0
D       10.1.1.0 [90/297372416] via 192.168.1.1, 00:01:04, Tunnel1
C    192.168.1.0/24 is directly connected, Tunnel1
S*   0.0.0.0/0 [1/0] via 192.1.20.5
```

Next hope is R1 (Hub) for reaching to all other spokes, but still packet will go directly to other spoke tunnel address instead of going via Hub.

```
R2#traceroute 10.3.3.3

Type escape sequence to abort.
Tracing the route to 10.3.3.3

  1 192.168.1.3 156 msec 156 msec 120 msec
```

```
R2#traceroute 10.4.4.4

Type escape sequence to abort.
Tracing the route to 10.4.4.4

  1 192.168.1.4 72 msec 44 msec 40 msec
```

Note: In DMVPN Phase 3, Using IP NHRP SHORTCUT and IP NHRP REDIRECT Commands, we are removing dependency from EIGRP to take decision for communication between spoke to spoke it's all based on NHRP.

THANK YOU !!!