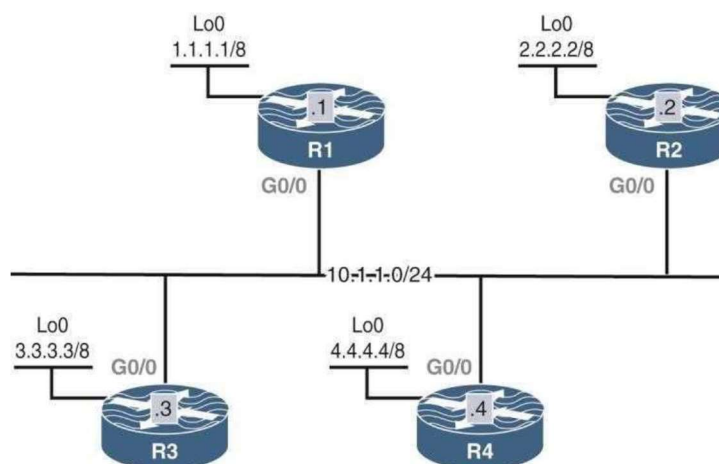


Lab 2

OSPF Broadcast Networks



This lab should be conducted on the Enterprise POD.

Lab Setup:

To copy and paste the initial configurations, go to the Initial-config folder ☐ OSPF folder ☐ Lab-2.

Task 1

Configure OSPF Area 0 on all routers and their directly connected interfaces. Ensure that Loopback0 interfaces are advertised with their correct mask. You should configure the OSPF router IDs to be 0.0.0.x, where x is the router number.

OSPF Network Types

One fact that is echoed throughout this section is that OSPF is a link-state routing protocol. It creates a graph of connected link states for all routers in the OSPF area domain. To create this graph, OSPF exchanges what are known as Link State Advertisements (LSAs) in the form of Link State Updates. These LSAs contain the descriptions of the router's links in the network and come in many types, each serving a specific purpose.

LSAs represent either a single link, a single network, or a single router.

Out of all the LSAs generated by the router, the most important is the Type-1 router LSA. This LSA is used to describe the router itself and its connected links. The Type-1 router LSA contains an individual link description for each physical or logical link that is enabled for OSPF on the local router. The way these links are described depends on the OSPF network type.

There are five main OSPF network types:

1. Broadcast
2. Non-Broadcast
3. Point-to-point
4. Point-to-multipoint
5. Point-to-multipoint non-broadcast

Each network type is explored in the next four labs beginning with the broadcast network type.

To start this section off, OSPF needs to be configured to run on all of the ethernet and loopback interfaces on all of the routers. The loopback interfaces, in particular, must advertise the proper subnet masks instead of the default OSPF /32 subnet mask used for loopback interfaces. The configuration below configures the proper router IDs, enables OSPF on the router interfaces, and ensures the loopbacks are advertised with proper subnet masks by changing the network type to point-to-point:

On R1:

```
R1(config)#router ospf 1
R1(config-router)#router-id 0.0.0.1
R1(config-router)#network 10.1.1.1 0.0.0.0 area 0
R1(config-router)#network 1.1.1.1 0.0.0.0 area 0

R1(config-router)#interface lo0
R1(config-if)#ip ospf network point-to-point
```

On R2:

```
R2(config)#router ospf 1
R2(config-router)#router-id 0.0.0.2
R2(config-router)#network 10.1.1.2 0.0.0.0 area 0
R2(config-router)#network 2.2.2.2 0.0.0.0 area 0

R2(config-router)#interface lo0
R2(config-if)#ip ospf network point-to-point
```

You should see the following console messages:

```
%OSPF-5-ADJCHG: Process 1, Nbr 0.0.0.1 on GigabitEthernet0/0 from LOADING
to FULL, Loading Done
```

On R3:

```
R3(config)#router ospf 1
R3(config-router)#router-id 0.0.0.3
R3(config-router)#network 10.1.1.3 0.0.0.0 area 0
R3(config-router)#network 3.3.3.3 0.0.0.0 area 0
```

```
R3(config-router)#interface lo0
R3(config-if)#ip ospf netw point-to-point
```

You should see the following console messages:

```
%OSPF-5-ADJCHG: Process 1, Nbr 0.0.0.1 on GigabitEthernet0/0 from LOADING
to FULL, Loading Done
```

```
%OSPF-5-ADJCHG: Process 1, Nbr 0.0.0.2 on GigabitEthernet0/0 from LOADING
to FULL, Loading Done
```

On R4:

```
R4(config)#router ospf 1
R4(config-router)#router-id 0.0.0.4
R4(config-router)#network 10.1.1.4 0.0.0.0 area 0
R4(config-router)#network 4.4.4.4 0.0.0.0 area 0
```

```
R4(config-router)#interface lo0
R4(config-if)#ip ospf network point-to-point
```

The **show ip route ospf | begin Gateway** command on all routers verifies the OSPF routes they learn:

On R1:

```
R1#show ip route ospf | begin Gate
Gateway of last resort is not set
```

```
O      2.0.0.0/8 [110/2] via 10.1.1.2, 00:30:23, GigabitEthernet0/0
O      3.0.0.0/8 [110/2] via 10.1.1.3, 00:29:51, GigabitEthernet0/0
O 4.0.0.0/8 [110/2] via 10.1.1.4, 00:29:22, GigabitEthernet0/0
```

On R2:

```
R2#show ip route ospf | begin Gate
```

```
Gateway of last resort is not set
```

```
O      1.0.0.0/8 [110/2] via 10.1.1.1, 00:31:49, GigabitEthernet0/0
O      3.0.0.0/8 [110/2] via 10.1.1.3, 00:31:22, GigabitEthernet0/0
O 4.0.0.0/8 [110/2] via 10.1.1.4, 00:30:54, GigabitEthernet0/0
```

On R3:

```
R3#show ip route ospf | begin Gate
```

```
Gateway of last resort is not set
```

```
O      1.0.0.0/8 [110/2] via 10.1.1.1, 00:31:47, GigabitEthernet0/0
O      2.0.0.0/8 [110/2] via 10.1.1.2, 00:31:47, GigabitEthernet0/0
O 4.0.0.0/8 [110/2] via 10.1.1.4, 00:31:16, GigabitEthernet0/0
```

On R4:

```
R4#show ip route ospf | begin Gate
```

```
Gateway of last resort is not set
```

```
O      1.0.0.0/8 [110/2] via 10.1.1.1, 00:31:44, GigabitEthernet0/0
O      2.0.0.0/8 [110/2] via 10.1.1.2, 00:31:44, GigabitEthernet0/0
O 3.0.0.0/8 [110/2] via 10.1.1.3, 00:31:44, GigabitEthernet0/0
```

To produce the output above, OSPF has to do some housekeeping on the back end. First and foremost, each router needs to know what kind of network to which its OSPF-enabled links are connected. With this information, it models the relationship between its link and the neighbors that are connected to that link.

In this case, each router's G0/0 interface is connected to a LAN segment. Such segments are called multiaccess or transit networks in OSPF terminology. It represents a link that can carry traffic that is not destined to a device on the link or connects to multiple other routers. Thus, the link is modeled as such in each router's Type-1 router LSA. Looking at the **show ip ospf database** command, reveals each router is advertising two links in their Type-1 router LSA as seen on R1:

On R1:

```
R1#show ip ospf database
```

```
OSPF Router with ID (0.0.0.1) (Process ID 1)
```

```
Router Link States (Area 0)
```

Link ID	ADV Router	Age	Seq#	Checksum	Link count
0.0.0.1	0.0.0.1	177	0x8000000A	0x009862	2
0.0.0.2	0.0.0.2	157	0x80000009	0x00AB4C	2
0.0.0.3	0.0.0.3	122	0x80000007	0x00C035	2
0.0.0.4	0.0.0.4	161	0x80000005	0x00D51E	2

Net Link States (Area 0)

Link ID	ADV Router	Age	Seq#	Checksum
10.1.1.2	0.0.0.2	157	0x80000004	0x00A074

The **show ip ospf database router self-originate** command on R1 goes into the details of its Type-1 router LSA. The LSA describes its link on the LAN segment as a “transit link” and the network connected to its loopback interface as a stub network:

R1#**show ip ospf database router self-originate**

OSPF Router with ID (0.0.0.1) (Process ID 1)

Router Link States (Area 0)

LS age: 278
Options: (No TOS-capability, DC)
LS Type: Router Links
Link State ID: 0.0.0.1
Advertising Router: 0.0.0.1
LS Seq Number: 8000000A
Checksum: 0x9862
Length: 48
Number of Links: 2

Link connected to: a Stub Network
(Link ID) Network/subnet number: 1.0.0.0
(Link Data) Network Mask: 255.0.0.0

Number of MTID metrics: 0
TOS 0 Metrics: 1

Link connected to: a Transit Network
(Link ID) Designated Router address: 10.1.1.2
(Link Data) Router Interface address: 10.1.1.1

Number of MTID metrics: 0
TOS 0 Metrics: 1

The link description for the transit area may look odd. It mentions a “Designated Router address” and a “Router Interface address”. Why does it do this? The simple answer is in an optimization OSPF performs for LAN segments.

In OSPF, LAN segments can be of two types, **broadcast** or **non-broadcast**. This speaks to the LAN’s ability to process broadcast/multicast packets. If the underlying LAN infrastructure can support broadcast traffic, then the router can multicast OSPF hello packets to discover neighbors. This is the **broadcast** network type that is the subject of this section.

On such LAN segments, it is possible to have multiple OSPF-speaking routers connected to the same segment. In case of the lab topology, there are four OSPF-speaking routers connected to the same LAN segment, the 10.1.1.0/24 LAN segment. ALL routers have full-mesh reachability to each other, meaning R1 can send a packet to R2, R3 and R4; R2 can send a packet to R1, R3 and, R4; and so on.

OSPF must model this full-mesh reachability in some way that makes sense and is efficient. One way, from R1’s perspective, is to model 3 point-to-point link adjacencies in its Type-1 LSA, one for its adjacency with each router. Each router on the segment would have to model the same 3 link adjacency leading to a grand total of 12 link adjacency descriptions in the OSPF database for a single LAN segment. If another OSPF-speaking router were added to the segment, it would increase to 20 link adjacencies. This model does not scale for large LAN or multi-access segments.

Instead, OSPF takes a different approach. It represents the LAN segment itself as a node on the graph and attaches all connected routers to that central node. This network node is described in the Type-2 network LSA. All routers connected to the LAN segment model a single link connected to the multiaccess segment as a transit network node. The following output truncates the the **show ip ospf database router self-originate** command on R1 and compares it to the **show ip ospf database network output**:

```
R1#show ip ospf database router self-originate | begin Transit
```

```
Link connected to: a Transit Network
(Link ID) Designated Router address: 10.1.1.2
(Link Data) Router Interface address: 10.1.1.1
Number of MTID metrics: 0
TOS 0 Metrics: 1
```

```
R1#show ip ospf database network
```

```
OSPF Router with ID (0.0.0.1) (Process ID 1)
```

```
Net Link States (Area 0)
```

```
LS age: 480
Options: (No TOS-capability, DC)
LS Type: Network Links
Link State ID: 10.1.1.2 (address of Designated Router)
Advertising Router: 0.0.0.2
```

```

LS Seq Number: 80000004
Checksum: 0xA074
Length: 40
Network Mask: /24
    Attached Router: 0.0.0.2
    Attached Router: 0.0.0.1
    Attached Router: 0.0.0.3
    Attached Router: 0.0.0.4

```

R1's Type-1 router LSA references what is known as a **Designated Router address** in its link description for the LAN segment. This designated router address is used as the identifier for the resulting Type-2 LSA. The Designated Router (DR) is used to solve another problem. Since all routers on the LAN segment must maintain the same information in their LSDBs, they must exchange LSAs with each other to achieve this synchronization. That can lead to repeated flooding of link-state information onto the LAN. Moreover, which router on the LAN should be the router that creates the network node for the LAN segment? The answer to this is the DR.

The DR is elected to help facilitate the LSA flooding and synchronization of Link-State information. All routers exchange LSAs with the DR and become fully adjacent, exchanging LSAs with the DR using the special "All- OSPF-DR" multicast group 224.0.0.6. The DR also creates the network node Type-2 network LSA shown above, adding the router IDs of all attached routers as well as the netmask for the LAN segment to the Type- 2 LSA. The Type-2 network LSA above represents the 10.1.1.0/24 network as indicated by the designated router/network mask combination. It also indicates that routers 0.0.0.1, 0.0.0.2, 0.0.0.3, and 0.0.0.4 are all connected to the 10.1.1.0/24 network node.

Organizing the LAN segment this way keeps link-state information flooding to a minimum. If a new router is added to the LAN segment, it becomes fully adjacent with the DR. The DR synchronizes its LSDB with the new router and updates the Type-2 LSA for the LAN segment with the new router's router ID. In the output above, the DR address is R2's IP address on the LAN segment. The DR is elected on the LAN segment using the following algorithm:

1. Highest priority (configured with the **ip ospf priority** command in interface configuration mode)
2. Highest OSPF router ID

This election is a one-time election. The DR is elected based on the exchange of hello packets during a given period of time which is roughly 40 seconds. Before carrying out DR elections, the routers wait for 40 seconds to see if there is a currently active DR on the segment. If not, the DR election is held. If the local router has not received a hello from another router on the segment with a higher OSPF priority or router ID, it elects itself as the DR and floods the Type-2 network LSA for the network segment.

If the DR fails, another router should take its place. However, to speed the convergence process, the router with the second highest priority or router-ID is elected as a Backup Designated Router (BDR). All routers

including the DR become fully adjacent with the BDR and exchange LSAs with the BDR. This way, when the DR fails, the BDR takes its place and floods a new Type-2 network LSA for the segment.

By default, all ethernet interfaces on Cisco IOS routers are designated as OSPF broadcast network types. This is because ethernet interfaces typically lead to ethernet LAN segments where multiple routers can be connected. The **show ip ospf interfaces G0/0** command on R1 reveals some very useful information regarding the **broadcast network type**:

```
R1#show ip ospf interface g0/0
```

```
GigabitEthernet0/0 is up, line protocol is up
  Internet Address 10.1.1.1/24, Area 0, Attached via Network Statement
  Process ID 1, Router ID 0.0.0.1, Network Type BROADCAST, Cost: 1
  Topology-MTID      Cost      Disabled      Shutdown      Topology Name
    0                1          no            no            Base
  Transmit Delay is 1 sec, State BDR, Priority 1
  Designated Router (ID) 0.0.0.2, Interface address 10.1.1.2
  Backup Designated router (ID) 0.0.0.1, Interface address 10.1.1.1
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    oob-resync timeout 40
    Hello due in 00:00:00
  Supports Link-local Signaling (LLS)
  Cisco NSF helper support enabled
  IETF NSF helper support enabled
  Index 1/2/2, flood queue length 0
  Next 0x0(0)/0x0(0)/0x0(0)
  Last flood scan length is 0, maximum is 1
  Last flood scan time is 0 msec, maximum is 0 msec
  Neighbor Count is 3, Adjacent neighbor count is 3
    Adjacent with neighbor 0.0.0.2 (Designated Router)
    Adjacent with neighbor 0.0.0.3
    Adjacent with neighbor 0.0.0.4
  Suppress hello for 0 neighbor(s)
```

The output above verifies the network type is **broadcast** on R1's G0/0. It also indicates the default hello and dead interval timers configured on the interface. By default, OSPF broadcast network type uses a hello interval of 10 seconds and a dead timer of 40 seconds. Throughout the next labs these settings will vary based on the different network types.

Finally, the OSPF broadcast network type installs routes into the routing table with the router that advertised the LSA on the LAN segment (to the DR first) as the next-hop as verified below:

```
R1#show ip route ospf | begin Gate
Gateway of last resort is not set
```

```
O    2.0.0.0/8 [110/2] via 10.1.1.2, 00:43:17, GigabitEthernet0/0
O    3.0.0.0/8 [110/2] via 10.1.1.3, 00:42:45, GigabitEthernet0/0
O    4.0.0.0/8 [110/2] via 10.1.1.4, 00:42:16, GigabitEthernet0/0
```

R1 installs the loopback 0 interface IP addresses of all routers with those routers as the next hop. This fact is very important for specialty partial-mesh cases such as DMVPN which is explained in the next section.

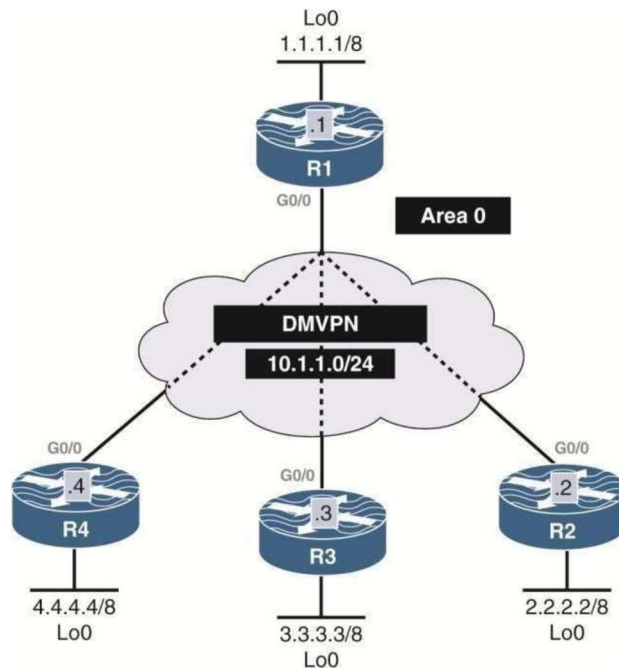
To review, the broadcast network type has the following characteristics:

1. Default network type on Ethernet interfaces
2. Hellos can be multicast to the 224.0.0.5 OSPF multicast address.
3. Hello and Dead timers are set to 10s and 40s respectively.
4. The next hop is the router that has the link to the route in the LSDB.
5. DR and BDR are elected to organize LSA exchange and generate the Type-2 LSA linking all routers on the segment together.
6. Routers exchange Link-State information with the DR/BDR using the 224.0.0.6 "All-OSPF-DR/BDR" multicast group.

Task 2

Lab Setup:

Erase the configuration of all routers and SW1. To copy and paste the initial configurations, go to the Initial-config folder ▢ OSPF folder ▢ Lab-2-Task-2.



Configure OSPF on the tunnel and Loopback0 interfaces of all routers, based on the following policy:

- R1 is the hub, and R2, R3, and R4 are configured as spokes. Do not change the topology.
- Configure the tunnel interfaces of all routers to be OSPF broadcast network type.
- The loopback interfaces should be advertised with their correct mask.
- Configure the router IDs of R1, R2, R3, and R4 to be 0.0.0.1, 0.0.0.2, 0.0.0.3, and 0.0.0.4, respectively.
- You should use static maps on the DMVPN network.
- Spokes should traverse the hub to reach all networks.

This lab examines some of the subtleties of the broadcast network type. Specifically, with how it behaves on partial-mesh NBMA networks such as DMVPN. To begin, OSPF should be configured according to the policy above. The following configures the tunnel and loopback0 interfaces for OSPF on each router. The router ID is set according to the task specifications using the **router-id** command in OSPF router configuration mode. OSPF is configured to advertise the actual mask on loopback interfaces by changing the network type on the loopback interfaces to point-to-point:

On All Routers:

```
Rx(config-router) #interface lo0
Rx(config-if) #ip ospf network point-to-point
```

On R1:

```
R1 (config) #router ospf 1
R1 (config-router) #router-id 0.0.0.1
R1 (config-router) #network 10.1.1.1 0.0.0.0 area 0
R1 (config-router) #network 1.1.1.1 0.0.0.0 area 0

R1 (config) #interface tunnel 1234
R1 (config-if) #ip ospf network broadcast
```

On R2:

```
R2 (config) #router ospf 1
R2 (config-router) #router-id 0.0.0.2
R2 (config-router) #network 2.2.2.2 0.0.0.0 area 0
R2 (config-router) #network 10.1.1.2 0.0.0.0 area 0

R2 (config) #interface tunnel 1234
R2 (config-if) #ip ospf network broadcast
```

On R3:

```
R3 (config) #router ospf 1
R3 (config-router) #router-id 0.0.0.3
R3 (config-router) #network 3.3.3.3 0.0.0.0 area 0
R3 (config-router) #network 10.1.1.3 0.0.0.0 area 0

R3 (config) #interface tunnel 1234
R3 (config-if) #ip ospf network broadcast
```

On R4:

```
R4 (config) #router ospf 1
R4 (config-router) #router-id 0.0.0.4

R4 (config-router) #network 4.4.4.4 0.0.0.0 area 0
R4 (config-router) #network 10.1.1.4 0.0.0.0 area 0

R4 (config) #interface tunnel 1234
R4 (config-if) #ip ospf network broadcast
```

After configuring all of the routers, there are no OSPF adjacency notifications on the routers. Furthermore, the **show ip ospf neighbor** command does not list any neighbors on any router:

On R1:

```
R1#show ip ospf neighbor  
R1#
```

On R2:

```
R2#show ip ospf neighbor  
R2#
```

On R3:

```
R3#show ip ospf neighbor  
R3#
```

On R4:

```
R4#show ip ospf neighbor  
R4#
```

Recall from the previous task that OSPF broadcast network type still uses multicast hello messages to discover neighbor adjacencies. However, on non-broadcast mediums such as DMVPN, these hello messages cannot be transmitted to all potential neighbors as multicast. To facilitate OSPF broadcast network type's multicast hello, the medium needs to support a form of pseudo-broadcasting. DMVPN supports this in the form of static NHRP multicast mappings. These mappings tell the router to which NBMA addresses multicast packets it originates should be forwarded.

The output of the **show ip nhrp multicast** command shows the current multicast mappings that exist on each router:

On R1:

```
R1#show ip nhrp multicast  
  
I/F      NBMA address
```

On R2:

```
R2#show ip nhrp multicast  
  
I/F      NBMA address
```

On R3:

```
R3#show ip nhrp multicast
```

```
  I/F      NBMA address
```

On R4:

```
R4#show ip nhrp multicast
```

```
  I/F      NBMA address
```

The output shows no multicast mappings exist on the routers. The cause of this can be seen using the **show run interface Tunnel1234 | begin interface** output:

On R1:

```
R1#show run interface tunnel 1234 | begin interface
```

```
interface Tunnel1234
 ip address 10.1.1.1 255.255.255.0
 no ip redirects
 ip nhrp map 10.1.1.2 192.1.2.2
 ip nhrp map 10.1.1.3 192.1.3.3
 ip nhrp map 10.1.1.4 192.1.4.4
 ip nhrp network-id 111
 ip ospf network broadcast
 tunnel source GigabitEthernet0/0
 tunnel mode gre multipoint
end
```

On R2:

```
R2#show run interface tunnel 1234 | begin interface
```

```
interface Tunnel1234

 ip address 10.1.1.2 255.255.255.0
 no ip redirects
 ip nhrp map 10.1.1.1 192.1.1.1
 ip nhrp network-id 222
 ip ospf network broadcast
 tunnel source GigabitEthernet0/0
 tunnel mode gre multipoint
end
```

On R3:

```
R3#show run interface tunnel 1234 | begin interface
```

```
interface Tunnel1234
 ip address 10.1.1.3 255.255.255.0
 no ip redirects
 ip nhrp map 10.1.1.1 192.1.1.1
 ip nhrp network-id 333
 ip ospf network broadcast
 tunnel source GigabitEthernet0/0
 tunnel mode gre multipoint
end
```

On R4:

```
R4#show run interface tunnel 1234 | begin interface
```

```
interface Tunnel1234
 ip address 10.1.1.4 255.255.255.0
 no ip redirects
 ip nhrp map 10.1.1.1 192.1.1.1
 ip nhrp network-id 444
 ip ospf network broadcast
 tunnel source GigabitEthernet0/0
 tunnel mode gre multipoint
end
```

The output contains no static or dynamic NHRP multicast mapping commands. To fix this, the static NHRP multicast mapping **command ip nhrp map multicast NBMA-address** should be added. R1 should have mappings for all four spoke NBMA addresses while the spokes require only a single mapping for R1's NBMA address. These mappings must be statically assigned out of compliance with the task requirements that only static mappings be configured. First the configuration is tested on R1 and R2:

On R1:

```
R1(config)#interface tunnnel 1234
R1(config-if)#ip nhrp map multicast 192.1.2.2
```

On R2:

```
R2(config)#interface tunnel 1234
R2(config-if)#ip nhrp map multicast 192.1.1.1
```

If both ends are not configured to map Multicast for each other's NBMA addresses, the OSPF adjacency will be established and torn down and you will get the following console message:

```
%OSPF-5-ADJCHG: Process 1, Nbr 0.0.0.2 on Tunnel1234 from LOADING to FULL, Loading Done
```

Let's verify the adjacency and see if these two routers are exchanging routes. As seen below, R1 learns the R2's loopback address via OSPF over the tunnel 1234 interface and the ping to the 2.2.2.2 address succeeds:

On R1:

```
R1#show ip ospf neighbor
```

Neighbor ID	Pri	State	Dead Time	Address	Interface
0.0.0.2	1	FULL/DR	00:00:35	10.1.1.2	Tunnel1234

```
R1#show ip route ospf | begin Gate
```

```
Gateway of last resort is not set
```

```
O      2.0.0.0/8 [110/1001] via 10.1.1.2, 00:03:26, Tunnel1234
```

```
R1#ping 2.2.2.2
```

```
Type escape sequence to abort.
```

```
Sending 5, 100-byte ICMP Echos to 2.2.2.2, timeout is 2 seconds:
```

```
!!!!
```

```
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/6 ms
```

Since the configuration was successful above it is repeated and applied to R3 and R4 as well:

```
R1(config)#interface tunnel 1234
```

```
R1(config-if)#ip nhrp map multicast 192.1.3.3
```

```
R1(config-if)#ip nhrp map multicast 192.1.4.4
```

On R3 & R4:

```
R3(config)#interface tunnel 1234
```

```
R3(config-if)#ip nhrp map multicast 192.1.1.1
```

Now the **show ip ospf neighbor** command on all routers verifies the OSPF adjacencies are up:

On R1:

```
R1#show ip ospf neighbor
```

Neighbor ID	Pri	State	Dead Time	Address	Interface
0.0.0.2	1	FULL/DROTHER	00:00:38	10.1.1.2	Tunnel1234
0.0.0.3	1	FULL/DROTHER	00:00:37	10.1.1.3	Tunnel1234

0.0.0.4	1	FULL/DR	00:00:32	10.1.1.4	Tunnel1234
---------	---	---------	----------	----------	------------

On R2:

R2#**show ip ospf neighbor**

Neighbor ID	Pri	State	Dead Time	Address	Interface
0.0.0.1	1	FULL/BDR	00:00:30	10.1.1.1	Tunnel1234

On R3:

R3#**show ip ospf neighbor**

Neighbor ID	Pri	State	Dead Time	Address	Interface
0.0.0.1	1	FULL/BDR	00:00:30	10.1.1.1	Tunnel1234

On R4:

R4#**show ip ospf neighbor**

Neighbor ID	Pri	State	Dead Time	Address	Interface
0.0.0.1	1	FULL/BDR	00:00:30	10.1.1.1	Tunnel1234

The routing tables on each router, however, tell a different story:

On R1:

R1#**show ip route ospf | begin Gate**

Gateway of last resort is not set

O 4.0.0.0/8 [110/1001] via 10.1.1.4, 00:05:19, Tunnel1234

On R2:

R2#**show ip route ospf | begin Gate** Gateway of last resort is not set

On R3:

R3#**show ip route ospf | begin Gate**

Gateway of last resort is not set

On R4:

R4#**show ip route ospf | begin Gate**

Gateway of last resort is not set

O 1.0.0.0/8 [110/1001] via 10.1.1.1, 00:05:29, Tunnel1234

Only R1 and R4 have routes in their routing tables and not even complete routing information. R2 and R3 have no OSPF-learned routes in their routing tables. What can be the cause of this? The answer is in the DR election process.

Recall from the previous task that OSPF broadcast network types elect a DR and BDR on each LAN segment. The DR creates a Type-2 network LSA to represent the LAN segment and advertises it to all its OSPF neighbors with the router IDs of all routers attached to the LSA. The routers only become fully adjacent with the DR and BDR and rely on these routers for their LSA exchange process.

The **show ip ospf interfaces brief** command output on each router reveals what role each router believes its tunnel interface to play on the LAN segment:

On R1:

```
R1#show ip ospf interface brief
```

Interface	PID	Area	IP Address/Mask	Cost	State	Nbrs	F/C
Lo0	1	0	1.1.1.1/8	1	P2P	0/0	
Tu1234	1	0	10.1.1.1/24	1000	BDR	3/3	

On R2:

```
R2#show ip ospf interface brief
```

Interface	PID	Area	IP Address/Mask	Cost	State	Nbrs	F/C
Lo0	1	0	2.2.2.2/8	1	P2P	0/0	
Tu1234	1	0	10.1.1.2/24	1000	DR	1/1	

On R3:

```
R3#show ip ospf interface brief
```

Interface	PID	Area	IP Address/Mask	Cost	State	Nbrs	F/C
Lo0	1	0	3.3.3.3/8	1	P2P	0/0	
Tu1234	1	0	10.1.1.3/24	1000	DR	1/1	

On R4:

```
R4#show ip ospf interface brief
```

Interface	PID	Area	IP Address/Mask	Cost	State	Nbrs	F/C
Lo0	1	0	4.4.4.4/8	1	P2P	0/0	
Tu1234	1	0	10.1.1.4/24	1000	DR	1/1	

In the above, R2, R3, and R4 all believe they are the DR on the LAN segment. This confusion results in three Type 2 LSAs being generated for the LAN segment as shown on R1:

On R1:

R1#**show ip ospf database network**

OSPF Router with ID (0.0.0.1) (Process ID 1)

Net Link States (Area 0)

LS age: 1099

Options: (No TOS-capability, DC)

LS Type: Network Links

Link State ID: 10.1.1.2 (address of Designated Router)

Advertising Router: 0.0.0.2

LS Seq Number: 80000001

Checksum: 0x81F

Length: 32

Network Mask: /24

Attached Router: 0.0.0.2

Attached Router: 0.0.0.1

LS age: 868

Options: (No TOS-capability, DC)

LS Type: Network Links

Link State ID: 10.1.1.3 (address of Designated Router)

Advertising Router: 0.0.0.3

LS Seq Number: 80000001

Checksum: 0x222

Length: 32

Network Mask: /24

Attached Router: 0.0.0.3

Attached Router: 0.0.0.1

LS age: 869

Options: (No TOS-capability, DC)

LS Type: Network Links

Link State ID: 10.1.1.4 (address of Designated Router)

Advertising Router: 0.0.0.4

LS Seq Number: 80000001

Checksum: 0xFB25

Length: 32

Network Mask: /24

Attached Router: 0.0.0.4

Attached Router: 0.0.0.1

This confusion in the LSDB causes the problems observed. The reason this confusion occurs is because R2, R3, and R4 cannot see each other's hello messages. The result is, they do not know that each other exists on the same network. Each is acting as if they are on an isolated network with R1 as the only other attached router on that network.

R1, on the other hand, only recognizes R4 as a DR as shown in the **show ip ospf neighbor** output below:

```
R1#show ip ospf neighbor
```

Neighbor ID	Pri	State	Dead Time	Address	Interface
0.0.0.2	1	FULL/DROTHER	00:00:34	10.1.1.2	Tunnel1234
0.0.0.3	1	FULL/DROTHER	00:00:35	10.1.1.3	Tunnel1234
0.0.0.4	1	FULL/DR	00:00:34	10.1.1.4	Tunnel1234

Because R1 believes R4 is the DR on the segment, it models its own link in its Type-1 LSA to the Type-2 network LSA generated by the DR R4, 10.1.1.4. as shown below:

```
R1#show ip ospf database router self-originate
```

```
OSPF Router with ID (0.0.0.1) (Process ID 1)
```

```
Router Link States (Area 0)
```

```
LS age: 1629
```

```
Options: (No TOS-capability, DC)
```

```
LS Type: Router Links
```

```
Link State ID: 0.0.0.1
```

```
Advertising Router: 0.0.0.1
```

```
LS Seq Number: 80000003
```

```
Checksum: 0x57BD
```

```
Length: 48
```

```
Number of Links: 2
```

```
Link connected to: a Stub Network
```

```
(Link ID) Network/subnet number: 1.0.0.0
```

```
(Link Data) Network Mask: 255.0.0.0
```

```
Number of MTID metrics: 0
```

```
TOS 0 Metrics: 1
```

```
Link connected to: a Transit Network
```

```
(Link ID) Designated Router address: 10.1.1.4
```

```
(Link Data) Router Interface address: 10.1.1.1
```

```
Number of MTID metrics: 0
```

```
TOS 0 Metrics: 1000
```

The consequence of this configuration is that even though R2 and R3 generate Type-2 network LSAs that list R1 as being connected to those network LSAs, R2 and R3 cannot install routes through R1 because R1 does not model a link to those network nodes in its Type-1 LSA. In OSPF, for a link to be valid, the Type-1 router LSA and the Type-2 LSA must model links to each other.

This is why R1's and R4's routes do not appear in R2's and R3's routing tables. Similarly, R2 and R3 do not model a link to R4's Type-2 network LSA, as R1 and R4 do. This is why R1 and R4 only have each other's routes in their routing tables. This point is proven by the output from R2, R3, and R4 below:

On R2:

```
R2#show ip ospf database router self-originate | begin Transit
```

```
Link connected to: a Transit Network
(Link ID) Designated Router address: 10.1.1.2
(Link Data) Router Interface address: 10.1.1.2
Number of MTID metrics: 0
TOS 0 Metrics: 1000
```

On R3:

```
R3#show ip ospf database router self-originate | begin Transit
```

```
Link connected to: a Transit Network
(Link ID) Designated Router address: 10.1.1.3
(Link Data) Router Interface address: 10.1.1.3
Number of MTID metrics: 0
TOS 0 Metrics: 1000
```

On R4:

```
R4#show ip ospf database router self-originate | begin Transit
```

```
Link connected to: a Transit Network
(Link ID) Designated Router address: 10.1.1.4
(Link Data) Router Interface address: 10.1.1.4
Number of MTID metrics: 0
TOS 0 Metrics: 1000
```

In the above, R2 and R3 model links to their own Type-2 LSA while R4 models a link to its own Type-2 LSA just like R1.

When configuring OSPF broadcast network types on an NBMA network such as DMVPN with partial mesh peerings as seen in this situation, it is important to consider the DR placement. The DR should be a router to which all routers connected to the NMBA network can exchange hellos with and form a full adjacency. All routers having only partial mesh neighborships should not be allowed to become the DR.

To prevent a router from becoming a DR, simply set its OSPF priority for that interface to zero. This setting disqualifies the router from ever becoming a DR or BDR. This configuration is applied using the **ip ospf priority 0** command in interface configuration mode and is performed on R2, R3, and R4 below:

On R2, R3, and R4:

```
Rx(config)#interface tunnel 1234
Rx(config-if)#ip ospf priority 0
```

On All Routers:

```
Rx#clear ip ospf process
Reset ALL OSPF processes? [no]: Yes
```

After configuration, all routers now agree that R1 should be the DR since it is the only router with a non- zero OSPF priority:

On R1:

```
R1#show ip ospf neighbor
```

Neighbor ID	Pri	State	Dead Time	Address	Interface
0.0.0.2	0	FULL/DROTHER	00:00:39	10.1.1.2	Tunnel1234
0.0.0.3	0	FULL/DROTHER	00:00:39	10.1.1.3	Tunnel1234
0.0.0.4	0	FULL/DROTHER	00:00:38	10.1.1.4	Tunnel1234

On R2:

```
R2#show ip ospf neighbor
```

Neighbor ID	Pri	State	Dead Time	Address	Interface
0.0.0.1	1	FULL/DR	00:00:32	10.1.1.1	Tunnel1234

On R3:

```
R3#show ip ospf neighbor
```

Neighbor ID	Pri	State	Dead Time	Address	Interface
0.0.0.1	1	FULL/DR	00:00:34	10.1.1.1	Tunnel1234

On R4:

```
R4#show ip ospf neighbor
```

Neighbor ID	Pri	State	Dead Time	Address	Interface
0.0.0.1	1	FULL/DR	00:00:31	10.1.1.1	Tunnel1234

The output on R1 confirms that R2, R3, and R4 are not acting as DR or BDR by the designation FULL/DROTHER. A router that is considered a DROTHER is neither a BDR nor DR on a LAN segment. The output from R2, R3, and R4 confirms that R1 is the DR on the segment. R1 floods the Type-2 LSA that all other routers point to in their Type-1 LSA:

On R1:

```
R1#show ip ospf database network | begin LS
```

```
LS age: 277
Options: (No TOS-capability, DC)
LS Type: Network Links
Link State ID: 10.1.1.1 (address of Designated Router)
Advertising Router: 0.0.0.1
LS Seq Number: 80000001
Checksum: 0xBA5F
Length: 40
Network Mask: /24
    Attached Router: 0.0.0.1
    Attached Router: 0.0.0.2
    Attached Router: 0.0.0.3
    Attached Router: 0.0.0.4
```

On R2:

```
R2#show ip ospf database router self-originate | begin Transit
```

```
Link connected to: a Transit Network
(Link ID) Designated Router address: 10.1.1.1
(Link Data) Router Interface address: 10.1.1.2
Number of MTID metrics: 0
TOS 0 Metrics: 1000
```

On R3:

```
R3#show ip ospf database router self-originate | begin Transit
```

```
Link connected to: a Transit Network
(Link ID) Designated Router address: 10.1.1.1
(Link Data) Router Interface address: 10.1.1.3
```

```
Number of MTID metrics: 0
TOS 0 Metrics: 1000
```

On R4:

```
R4#show ip ospf database router self-originate | begin Transit
```

```
Link connected to: a Transit Network
(Link ID) Designated Router address: 10.1.1.1
(Link Data) Router Interface address: 10.1.1.4
Number of MTID metrics: 0
TOS 0 Metrics: 1000
```

Now that the LSDB is in agreement on all routers, they have the proper routes in their routing table to reach all attached networks over the DMVPN:

On R1:

```
R1#show ip route ospf | begin Gate
```

```
Gateway of last resort is not set
```

```
O    2.0.0.0/8 [110/1001] via 10.1.1.2, 00:11:35, Tunnel1234
O    3.0.0.0/8 [110/1001] via 10.1.1.3, 00:11:35, Tunnel1234
O    4.0.0.0/8 [110/1001] via 10.1.1.4, 00:11:35, Tunnel1234
```

On R2:

```
R2#show ip route ospf | begin Gate
```

```
Gateway of last resort is not set
```

```
O    1.0.0.0/8 [110/1001] via 10.1.1.1, 00:12:51, Tunnel1234
O    3.0.0.0/8 [110/1001] via 10.1.1.3, 00:12:41, Tunnel1234
O    4.0.0.0/8 [110/1001] via 10.1.1.4, 00:12:41, Tunnel1234
```

On R3:

```
R3#show ip route ospf | begin Gate
```

```
Gateway of last resort is not set
```

```
O    1.0.0.0/8 [110/1001] via 10.1.1.1, 00:13:18, Tunnel1234
O    2.0.0.0/8 [110/1001] via 10.1.1.2, 00:13:18, Tunnel1234
O    4.0.0.0/8 [110/1001] via 10.1.1.4, 00:13:08, Tunnel1234
```

On R4:

```
R4#show ip route ospf | begin Gate
```

```
Gateway of last resort is not set
```

```
O      1.0.0.0/8 [110/1001] via 10.1.1.1, 00:13:53, Tunnel1234
O      2.0.0.0/8 [110/1001] via 10.1.1.2, 00:13:53, Tunnel1234
O 3.0.0.0/8 [110/1001] via 10.1.1.3, 00:13:53, Tunnel1234
```

Even though all routers possess all routes to networks reachable over the DMVPN, it doesn't necessarily mean they have the full reachability to those routes. Take R2 for example:

On R2:

```
R2#ping 1.1.1.1
```

```
Type escape sequence to abort.
```

```
Sending 5, 100-byte ICMP Echos to 1.1.1.1, timeout is 2 seconds:
```

```
!!!!
```

```
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/2 ms
```

```
R2#ping 3.3.3.3
```

```
Type escape sequence to abort.
```

```
Sending 5, 100-byte ICMP Echos to 3.3.3.3, timeout is 2 seconds:
```

```
.....
```

```
Success rate is 0 percent (0/5)
```

```
R2#ping 4.4.4.4
```

```
Type escape sequence to abort.
```

```
Sending 5, 100-byte ICMP Echos to 4.4.4.4, timeout is 2 seconds:
```

```
.....
```

```
Success rate is 0 percent (0/5)
```

R2 cannot reach R3 and R4's loopback interfaces. The reason behind this is a result of another aspect of the broadcast network type. Broadcast network type assumes all routers can reach all routers over the multi-access network. As such, when routers install routes to reach networks through the broadcast network, the next-hop installed is the next-hop of the router that owns the prefix. In this case, the next-hop for 1.1.1.1 is R1, for 2.2.2.2 is R2, for 3.3.3.3 is R3, and for 4.4.4.4 is R4.

R2 does not have reachability to the overlay addresses of R3 and R4 because there is no underlying DMVPN mapping made proven by the above ping output on R2.

Without this underlying mapping, R2 cannot reach those networks. To fix this, static NHRP mappings need to be made on R2 to achieve reachability. This mapping maps R3's overlay address 10.1.1.3 to R1's NBMA address 192.1.1.1 to maintain hub-and-spoke traffic flow on R2 in compliance with the task objectives.

The configuration is performed using the **ip nhrp map 10.1.1.3 192.168.1.1** command in interface configuration mode for the DMVPN tunnel interface on R2. R3 also needs a mapping command for R2's overlay address using the **ip nhrp map 10.1.1.2 192.168.1.1** command in interface configuration mode for the DMVPN tunnel interface.

On R2:

```
R2(config)#interface tunnel 1234
R2(config-if)#ip nhrp map 10.1.1.3 192.1.1.1
```

On R3:

```
R3(config)#interface tunnel 1234
R3(config-if)#ip nhrp map 10.1.1.2 192.1.1.1
```

A ping between R2 and R3's overlay addresses confirms the mapping is working properly:

```
R2#ping 10.1.1.3
```

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 10.1.1.3, timeout is 2 seconds:

!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/2 ms

Since the configuration was a success, it is applied to the remaining spokes to create full-mesh hub-and-spoke logical mappings between the spoke routers.

On R2:

```
R2(config)#interface tunnel 1234
R2(config-if)#ip nhrp map 10.1.1.4 192.1.1.1
```

On R3:

```
R3(config)#interface tunnel 1234
R3(config-if)#ip nhrp map 10.1.1.4 192.1.1.1
```

On R4:

```
R4(config)#interface tunnel 1234
R4(config-if)#ip nhrp map 10.1.1.2 192.1.1.1
```

```
R4(config-if)#ip nhrp map 10.1.1.3 192.1.1.1
```

To confirm the mappings the **show ip nhrp** command is used to verify the addresses are mapped properly:

On R2:

```
R2#show ip nhrp
```

```
10.1.1.1/32 via 10.1.1.1
  Tunnel1234 created 01:14:54, never expire
  Type: static, Flags: used
  NBMA address: 192.1.1.1
10.1.1.3/32 via 10.1.1.3
  Tunnel1234 created 00:11:10, never expire
  Type: static, Flags: used
  NBMA address: 192.1.1.1
10.1.1.4/32 via 10.1.1.4
  Tunnel1234 created 00:06:19, never expire
  Type: static, Flags:
  NBMA address: 192.1.1.1
```

On R3:

```
R3#show ip nhrp
```

```
10.1.1.1 /32 via 10.1.1.1
  Tunnel1234 created 01:15:27, never expire
  Type: static, Flags: used
  NBMA address: 192.1.1.1
10.1.1.2 /32 via 10.1.1.2
  Tunnel1234 created 00:11:06, never expire
  Type: static, Flags: used
  NBMA address: 192.1.1.1
10.1.1.4/32 via 10.1.1.4
  Tunnel1234 created 00:06:22, never expire
  Type: static, Flags:
  NBMA address: 192.1.1.1
```

On R4:

```
R4#show ip nhrp
```

```
10.1.1.1 /32 via 10.1.1.1
  Tunnel1234 created 01:15:46, never expire
```

Type: static, Flags: used

```

NBMA address: 192.1.1.1
10.1.1.2 /32 via 10.1.1.2
Tunnel1234 created 00:06:18, never expire
Type: static, Flags:
NBMA address: 192.1.1.1
10.1.1.3 /32 via 10.1.1.3
Tunnel1234 created 00:06:10, never expire
Type: static, Flags:
NBMA address: 192.1.1.1

```

Using R2 as a test subject, pings are executed to the networks behind the spokes R3 and R4. A trace route to

3.3.3.3 also confirms traffic is flowing to the R1 the hub first and then to R3:

On R2:

R2#**ping 3.3.3.3**

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 3.3.3.3, timeout is 2 seconds:

!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/3 ms

R2#**ping 4.4.4.4**

Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 4.4.4.4, timeout is 2 seconds:

!!!!

Success rate is 100 percent (5/5), round-trip min/avg/max = 2/2/2 ms

R2#**traceroute 3.3.3.3 numeric**

Type escape sequence to abort.

Tracing the route to 3.3.3.3

VRF info: (vrf in name/id, vrf out name/id)

```

 1 10.1.1.1 2 msec 1 msec 1 msec

```

```

 2 10.1.1.3 2 msec * 2 msec

```

This section covered the subtleties of the broadcast network type through its operation over NBMA networks such as DMVPN. The following summarizes these subtleties:

1. Multicast capability needs to be provided in some form by the NBMA network configuration

2. The DR of the NBMA network needs to be a router that has full-mesh OSPF adjacencies with all routers in the topology. Any router lacking this should be configured to be ineligible to be a DR using the **ip ospf priority 0** interface-level command.
3. In partial-mesh hub-and-spoke NBMA networks, the spokes must have mappings to reach remote spoke next-hop addresses.

Task 3

Erase the startup configuration of the routers, the config.text and the VLAN.dat of the switches and reload them before proceeding to the next lab.