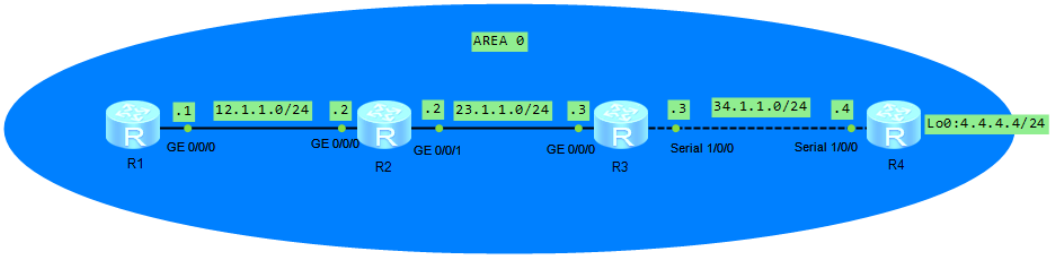
【HCIP 实验 06】OSPF-1 邻居状态机和 DR 选举

一、 实验拓扑



二、 实验需求及解法

通过本次实验，验证 OSPF 邻居状态机变化过程，以及 DR 选举过程。

1.如图所示，配置设备 IP 地址。

解法略

2.配置 OSPF

2.1 手动设置各设备 RID，如下表：

R1:1.1.1.1

R2:2.2.2.2

R3:3.3.3.3

R4:4.4.4.4

2.2 所有通配符都使用 0.0.0.0 精确通告。

2.3 所有接口都划入区域 0

R1：

```
ospf 1 router-id 1.1.1.1
 area 0.0.0.0
  network 12.1.1.1 0.0.0.0
#
```

R2：

```
ospf 1 router-id 2.2.2.2
 area 0.0.0.0
  network 12.1.1.2 0.0.0.0
```

```
   network 23.1.1.2 0.0.0.0
#
R3：
ospf 1 router-id 3.3.3.3
 area 0.0.0.0
   network 23.1.1.3 0.0.0.0
   network 34.1.1.3 0.0.0.0
#
R4:
ospf 1 router-id 4.4.4.4
 area 0.0.0.0
   network 4.4.4.4 0.0.0.0
   network 34.1.1.4 0.0.0.0
```

2.4 R1 与 R2 之间配置 R2 的 DR 优先级为 0，强制选择 R1 为 DR。

```
R2:
interface GigabitEthernet0/0/0
 ospf dr-priority 0
```

2.5 R2 的 G0/0/1 接口的 cost 修改为 100，R3 的 G0/0/0 接口的 cost 修改为 200。观察 R3 和 R2 收到 4.4.4.4/32 路由的 cost。

```
R2：
interface GigabitEthernet0/0/1
 ospf cost 100
R3:
interface GigabitEthernet0/0/0
 ospf cost 200
#
[R3]dis ip routing-table
```

```
Destination/Mask    Proto   Pre  Cost      Flags NextHop
        4.4.4.4/32  OSPF    10   48          D   34.1.1.4
```

```
[R2]dis ip routing-table
```

```
Destination/Mask    Proto   Pre  Cost      Flags NextHop
        4.4.4.4/32  OSPF    10   148         D   23.1.1.3
```

R3 看到 cost 为 48，R2 看到 cost 为 148。

说明 ospf 累加 cost 只计算去往该目标流量的出接口 cost。

2.6 R3 与 R4 之间修改 hello 时间为 5s。

R3：
interface Serial1/0/0
 ospf timer hello 5
#
R4:
interface Serial1/0/0
 ospf timer hello 5

2.7 修改 R4 的 Loopback0 网络类型为 broadcast, 在 R1/2/3 观察收到路由条目的变化情况。
R4:
interface LoopBack0
 ip address 4.4.4.4 255.255.255.0
 ospf network-type broadcast
#
[R2]dis ip routing-table

Destination/Mask	Proto	Pre	Cost	Flags	NextHop
4.4.4.0/24	OSPF	10	148	D	23.1.1.3

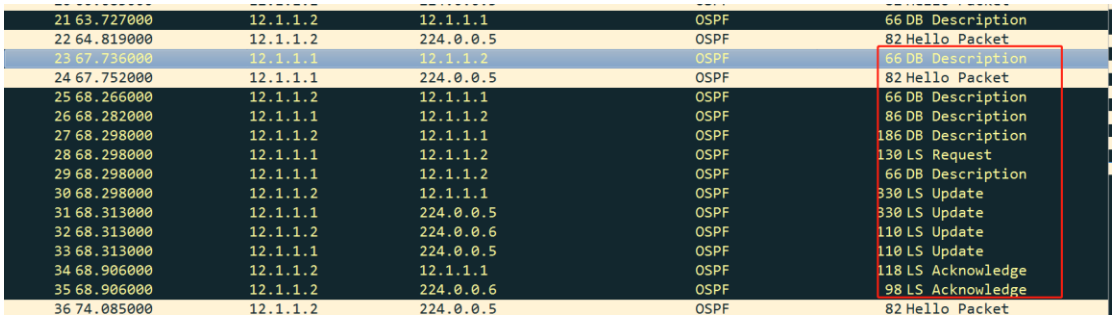
R2 收到的路由变成了与实际接口配置相同的 24 位。
（默认情况下，环回口生成的路由都是 32 位。）

3. 按照以下步骤观察 R1 与 R2 的邻居关系建立过程
3.1 R1 上使用命令 debugging ospf packet
3.2 在 R1 与 R2 之间开启抓包。
3.3 R1 重启 ospf 进程，reset ospf process。
（见下页）

```
<R1>reset ospf process
Warning: The OSPF process will be reset. Continue? [Y/N]:y
Dec  4 2019 16:30:38-08:00 R1 %%01OSPF/3/NBR_CHG_DOWN(l)[0]:Neighbor event:neigh
bor state changed to Down. (ProcessId=256, NeighborAddress=2.2.2.2, NeighborEven
t=KillNbr, NeighborPreviousState=Full, NeighborCurrentState=Down)
<R1>
<R1>
Dec  4 2019 16:30:38-08:00 R1 %%01OSPF/3/NBR_DOWN_REASON(l)[1]:Neighbor state le
aves full or changed to Down. (ProcessId=256, NeighborRouterId=2.2.2.2, Neighbor
AreaId=0, NeighborInterface=GigabitEthernet0/0/0,NeighborDownImmediate reason=Ne
ighbor Down Due to Kill Neighbor, NeighborDownPrimeReason=OSPF Process Reset Ne
ighborChangeTime=2019-12-04 16:30:38-08:00)
<R1>
Dec  4 2019 16:30:39-08:00 R1 %%01OSPF/4/NBR_CHANGE_E(l)[2]:Neighbor changes eve
nt: neighbor status changed. (ProcessId=256, NeighborAddress=2.1.1.12, NeighborE
vent=HelloReceived, NeighborPreviousState=Down, NeighborCurrentState=Init)
<R1>
Dec  4 2019 16:30:39-08:00 R1 %%01OSPF/4/NBR_CHANGE_E(l)[3]:Neighbor changes eve
nt: neighbor status changed. (ProcessId=256, NeighborAddress=2.1.1.12, NeighborE
vent=2WayReceived, NeighborPreviousState=Init, NeighborCurrentState=2Way)
<R1>
Dec  4 2019 16:31:17-08:00 R1 %%01OSPF/4/NBR_CHANGE_E(l)[4]:Neighbor changes eve
nt: neighbor status changed. (ProcessId=256, NeighborAddress=2.1.1.12, NeighborE
vent=AdjOk?, NeighborPreviousState=2Way, NeighborCurrentState=ExStart)
<R1>
Dec  4 2019 16:31:20-08:00 R1 %%01OSPF/4/NBR_CHANGE_E(l)[5]:Neighbor changes eve
nt: neighbor status changed. (ProcessId=256, NeighborAddress=2.1.1.12, NeighborE
vent=NegotiationDone, NeighborPreviousState=ExStart, NeighborCurrentState=Exchan
ge)
<R1>
Dec  4 2019 16:31:20-08:00 R1 %%01OSPF/4/NBR_CHANGE_E(l)[6]:Neighbor changes eve
nt: neighbor status changed. (ProcessId=256, NeighborAddress=2.1.1.12, NeighborE
vent=ExchangeDone, NeighborPreviousState=Exchange, NeighborCurrentState=Loading)

<R1>
Dec  4 2019 16:31:20-08:00 R1 %%01OSPF/4/NBR_CHANGE_E(l)[7]:Neighbor changes eve
nt: neighbor status changed. (ProcessId=256, NeighborAddress=2.1.1.12, NeighborE
vent=LoadingDone, NeighborPreviousState=Loading, NeighborCurrentState=Full)
<R1>
```

上图显示了 R1 的 ospf 进程重启后，邻接关系的建立过程。



21	63.727000	12.1.1.2	12.1.1.1	OSPF	66 DB Description
22	64.819000	12.1.1.2	224.0.0.5	OSPF	82 Hello Packet
23	67.736000	12.1.1.1	12.1.1.2	OSPF	66 DB Description
24	67.752000	12.1.1.1	224.0.0.5	OSPF	82 Hello Packet
25	68.266000	12.1.1.2	12.1.1.1	OSPF	66 DB Description
26	68.282000	12.1.1.1	12.1.1.2	OSPF	86 DB Description
27	68.298000	12.1.1.2	12.1.1.1	OSPF	186 DB Description
28	68.298000	12.1.1.1	12.1.1.2	OSPF	130 LS Request
29	68.298000	12.1.1.1	12.1.1.2	OSPF	66 DB Description
30	68.298000	12.1.1.2	12.1.1.1	OSPF	330 LS Update
31	68.313000	12.1.1.1	224.0.0.5	OSPF	330 LS Update
32	68.313000	12.1.1.2	224.0.0.6	OSPF	110 LS Update
33	68.313000	12.1.1.1	224.0.0.5	OSPF	110 LS Update
34	68.906000	12.1.1.2	12.1.1.1	OSPF	118 LS Acknowledge
35	68.906000	12.1.1.2	224.0.0.6	OSPF	98 LS Acknowledge
36	74.085000	12.1.1.2	224.0.0.5	OSPF	82 Hello Packet

抓包可以看到 hello DD LSR LSU LSAck 五种报文的交互。