

实验：IS-IS

HCIP 分解实验 - IS-IS

臧家林制作



IS-IS 实验 1：IS-IS 基本配置

IS-IS 实验 2：IS-IS 邻接关系

IS-IS 实验 3：IS-IS 链路状态数据库

IS-IS 实验 4：IS-IS DIS

IS-IS 实验 5：IS-IS 开销值和协议优先级

IS-IS 实验 6：IS-IS 路由聚合

IS-IS 实验 7：IS-IS 缺省路由

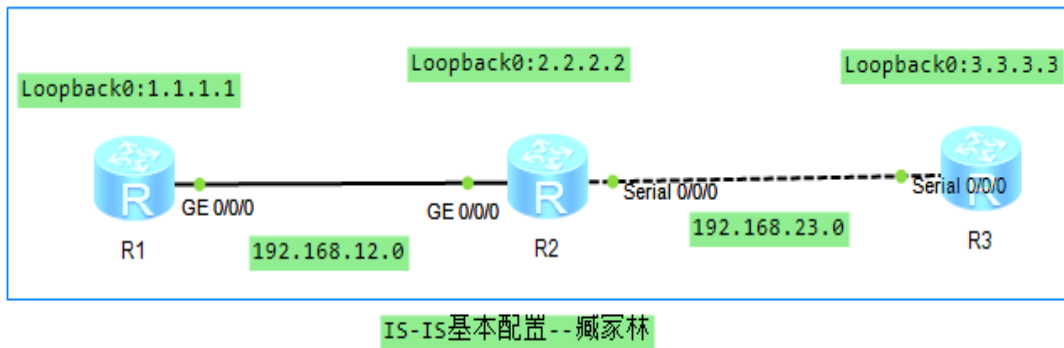
IS-IS 实验 8：IS-IS 路由引入

IS-IS 实验 9：IS-IS 路由过滤

IS-IS 实验 10：IS-IS 路由渗透（泄漏）

=====

IS-IS 实验 1：IS-IS 基本配置



3 个网络实体名， padding ， PPP 的 3-way ， mtu ， IP 地址不同建立邻居， 认证

基本配置

R1:

```
undo ter mo
sys
sysname R1
user-interface console 0
idle-timeout 0 0
int loop 0
ip add 1.1.1.1 24
int g0/0/0
ip add 192.168.12.1 24
q
```

R2:

```
undo ter mo
sys
sysname R2
user-interface console 0
idle-timeout 0 0
int loop 0
ip add 2.2.2.2 24
int g0/0/0
ip add 192.168.12.2 24
int s0/0/0
```

```
ip add 192.168.23.2 24
q
```

```
R3:
undo ter mo
sys
sysname R3
user-interface console 0
idle-timeout 0 0
int loop 0
ip add 3.3.3.3 24
int s0/0/0
ip add 192.168.23.3 24
q
```

配置完成后，R2 ping 一下直连，测试连通性

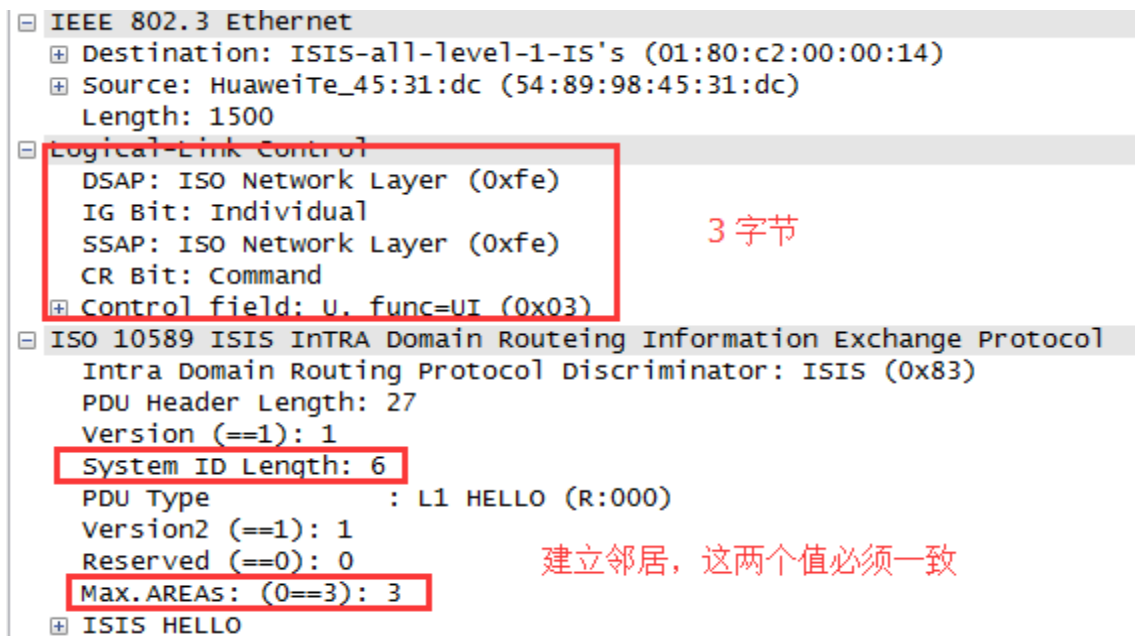
配置 IS-IS 路由协议

```
R1:
isis
network-entity 10.0000.0000.0001.00
int g0/0/0
isis enable
int loo 0
isis enable
```

默认 IS-IS 进程号为 1，配置路由器的网络实体名，即指定系统的区域 ID 和系统 ID

路由器上需要运行 IS-IS 的接口必须使用 isis enable 命令来启用 IS-IS 协议

抓取 ISIS hello 报文，



用 IS-IS 协议 物理接口的 MTU 为 1497，有 3 个字节给 LLC 使用了

system ID length 默认为 6，是系统 ID 号长度

max area 默认为 3，代表最大支持 3 个 network-entity

每台设备最多可以配置三个 NET，系统 ID 必须是相同，区域 ID 可以不同。

一般情况下，一个路由只需要配置一个区域地址，且同一区域中所有节点的区域地址都要相同。为了支持区域的平滑合并，分割及转换，在设备的实现中，一个 ISIS 进程下最多可配置 3 个区域地址。

```
isis 1
```

```
network-entity 10.0000.0000.0001.00
```


```
network-entity 20.0000.0000.0001.00
```

```
network-entity 30.0000.0000.0001.00
```

```
[R1]dis isis int
```

Interface information for ISIS(1)

Interface	Id	IPV4.State	IPV6.State	MTU	Type
GE0/0/0	001	Up	Down	1497	L1/L2
Loop0	001	Up	Down	1500	L1/L2



hello 报文中只有 holding timer , $\text{hello} * 3 = \text{holding timer}$

hello 时间不同 , 不影响建立邻居

但 MTU 要相同

```
IS-IS HELLO
  Circuit type           : Level 1 and 2, reserved(0x00 == 0)
  System-ID {Sender of PDU} : 0000.0000.0001
  Holding timer: 9
  PDU length: 1497
  Priority                : 64, reserved(0x00 == 0)
  System-ID {Designated IS} : 0000.0000.0001.01
```

hello 报文中的 padding 在 P2P 网络中 , 建立好之后 , 就没有了 , 节约带宽 , 在广播网络中一直有

int s0/0/0

isis padding-hello 一直有 padding

int s0/0/0

isis small-hello 没有 padding

两条命令互斥

isis 作为链路状态协议 , 一个重要的原则是要保证整个网络中

所有设备的 lsdb 数据库保持完全一致，这样才能保证各设备计算出的最小生成树 (spf) 是基于同一个拓扑，从而保证不会出现环路。isis 报文作为链路层报文，发送和接收过程是没有分片的概念的，也就是说接口的 mtu 决定了可以收到的 lsp 的最大长度。

举个例子，a 与 b 建立 isis 邻居。但 a 的接口 mtu 为 1000，b 的接口 mtu 为 2000，b 的 lsdb 中有一条长度为 1500 的 lsp。在这种情况下，这条 lsp 发到 a 是要丢弃的，因为超过了 a 的接口 mtu。结果是 a 与 b 的 lsdb 就不一致了。

解决方案就是把 hello 报文填充到接口的 mtu 长度。b 的 hello 报文全部长度现在是 2000，发到 a 就丢弃了。建不起来 isis 邻居，自然不会有不一致的现象。

建不起来邻居可以识别出来继而修改不一致的 mtu，而数据库不一致导致的后果完全无法预见，是无法接受的。

华为设备的实现：广播网的 hello 是根据 mtu 填充的；对于 p2p 的 hello，华为是根据 ls plength (单条 lsp 的最大生成长度) 的大小进行填充。在 p2p 邻居 up 后，不再对 hello 报文进行填充。当然，也可以通过 small-hello 和 padding-hello 命令强制指定是否填充。

```

- - - - -
[ ] ISIS HELLO
    Circuit type           : Level 1 and 2, reserved(0x00 == 0)
    System-ID {Sender of PDU} : 0000.0000.0001
    Holding timer: 9
    PDU length: 1497
    Priority                : 64, reserved(0x00 == 0)
    System-ID {Designated IS} : 0000.0000.0001.01
    [x] Area address(es) (6)
    [x] IS Neighbor(s) (6)
    [x] IP Interface address(es) (4)
    [x] Protocols supported (1)
    [x] Restart signaling (3)
    [x] Multi Topology (2)
    [x] Padding (255)
    [x] Padding (255)
    [x] Padding (255)
    [x] Padding (255)
    [x] Padding (255)
    [x] Padding (149)

```

```

R2:
isis
network-entity 10.0000.0000.0002.00
int g0/0/0
isis enable
int s0/0/0
isis enable
int loo 0
isis enable

```

```

R3:
isis
network-entity 10.0000.0000.0003.00
int s0/0/0
isis enable
int loo 0
isis enable

```

P2P 链路可以用 2-way 也可以用 3-way 来建立

```
int s0/0/0
isis ppp-negotiation 3-way (only)
```

不在同一个网段，建立 ISIS 邻居

```
R2 :
int s0/0/0
isis peer-ip-ignore
```

```
R3 :
int s0/0/0
ip add 192.168.35.3 24
isis peer-ip-ignore
```

设置了 peer-ip-ignore 后，P2P 网络不是同一个网段的两边，也能建立邻居

配置完成后在 R2 上查看邻居信息 display isis peer

System Id	Interface	Circuit Id	State	HoldTime	Type
0001.0000.0001	GE0/0/0	0001.0000.0001.01	Up	8s	L1(L1L2)
0001.0000.0001	GE0/0/0	0001.0000.0001.01	Up	7s	L2(L1L2)
0001.0000.0003	S0/0/0	0000000001	Up	29s	L1L2

Total Peer(s): 3

与系统 ID 为 0000.0000.0001 的邻居建立了 Level-1 邻接关系及 Level-2 邻接关系，3 条邻居信息的状态均为 up，表示 IS-IS 邻接关系已经正常建立。

=====

由于系统 ID 不易于管理和维护时的识别和认读，可以在 IS-IS 视图下设置为动态主机名。

```
R1:
isis
is-name R1
```

```
R2:
isis
is-name R2
```

```
R3:
isis
is-name R3
```

查看一下，系统 ID 已经被动态主机名替换

```
[R2]dis isis peer
```

Peer information for ISIS(1)

System Id	Interface	Circuit Id	State	HoldTime	Type	PRI

R1	GE0/0/0	R1.01	Up	9s	L1(L1L2)	64
R1	GE0/0/0	R1.01	Up	8s	L2(L1L2)	64
R3	S0/0/0	0000000001	Up	24s	L1L2	--

查看 R1 的路由表

```
<R1>display ip routing-table
```

1.1.1.0/24	Direct	0	0	D	1.1.1.1
1.1.1.1/32	Direct	0	0	D	127.0.0.1
2.2.2.0/24	ISIS-L1	15	10	D	192.168.12.2
3.3.3.0/24	ISIS-L1	15	20	D	192.168.12.2
127.0.0.0/8	Direct	0	0	D	127.0.0.1
127.0.0.1/32	Direct	0	0	D	127.0.0.1
192.168.12.0/24	Direct	0	0	D	192.168.12.1
192.168.12.1/32	Direct	0	0	D	127.0.0.1
192.168.23.0/24	ISIS-L1	15	20	D	192.168.12.2

可以看到，R1 已经获得了其他路由器的 Loopback 0 接口的路由，以及其他非直连网段的路由，优先级为 15

=====

配置 IS-IS 认证功能

接口认证：对 Level-1 和 Level-2 的 Hello 报文进行认证

区域认证：对 Level-1 的 SNP 和 LSP 报文进行认证

路由域认证：对 Level-2 的 SNP 和 LSP 报文进行认证

认证方式：Null、明文、MD5

接口认证

R1：

```
int g0/0/0
```

```
isis authentication-mode md5 huawei
```

只做一边的认证，邻居关系断开

<R1>

IS-IS邻接关系--臧冢林

基本配置

R1:

```
undo ter mo
sys
sysname R1
user-interface console 0
idle-timeout 0 0
int loop 0
ip add 1.1.1.1 24
int g0/0/0
ip add 192.168.12.1 24
int g0/0/1
ip add 192.168.13.1 24
q
```

R2:

```
undo ter mo
sys
sysname R2
user-interface console 0
idle-timeout 0 0
int loop 0
ip add 2.2.2.2 24
int g0/0/0
ip add 192.168.12.2 24
int g0/0/2
ip add 192.168.23.2 24
q
```

R3:

```
undo ter mo
sys
sysname R3
```

```
user-interface console 0
idle-timeout 0 0
int g0/0/1
ip add 192.168.13.3 24
int g0/0/2
ip add 192.168.23.3 24
int g0/0/0
ip add 192.168.34.3 24
q
```

```
R4:
undo ter mo
sys
sysname R4
user-interface console 0
idle-timeout 0 0
int g0/0/0
ip add 192.168.34.4 24
int g0/0/1
ip add 192.168.45.4 24
q
```

```
R5:
undo ter mo
sys
sysname R5
user-interface console 0
idle-timeout 0 0
int g0/0/1
ip add 192.168.45.5 24
int g0/0/0
ip add 192.168.56.5 24
q
```

```
R6:
undo ter mo
sys
sysname R6
user-interface console 0
idle-timeout 0 0
int loop 0
ip add 6.6.6.6 24
int g0/0/0
ip add 192.168.56.6 24
q
```

配置 IS-IS 协议

```
R1:
isis
network-entity 10.0000.0000.0001.00
is-name R1
int LoopBack 0
isis enable
int g0/0/0
isis enable
int g0/0/1
isis enable
```

```
R2:
isis
network-entity 10.0000.0000.0002.00
is-name R2
int LoopBack 0
isis enable
int g0/0/0
isis enable
int g0/0/2
isis enable
```

R3:

isis

network-entity 10.0000.0000.0003.00

is-name R3

int g0/0/0

isis enable

int g0/0/1

isis enable

int g0/0/2

isis enable

R4:

isis

network-entity 20.0000.0000.0004.00

is-name R4

int g0/0/0

isis enable

int g0/0/1

isis enable

R5:

isis

network-entity 30.0000.0000.0005.00

is-name R5

int g0/0/0

isis enable

int g0/0/1

isis enable

R6:

isis

network-entity 30.0000.0000.0006.00

is-name R6

```
int LoopBack 0
isis enable
int g0/0/0
isis enable
```

配置完成后，测试 R1 R6 的连通性

```
<R1>ping -c 3 -a 1.1.1.1 6.6.6.6
```

```
<R1>ping -c 3 -a 1.1.1.1 6.6.6.6
```

```
PING 6.6.6.6: 56 data bytes, press CTRL_C to break
```

```
Reply from 6.6.6.6: bytes=56 Sequence=1 ttl=252 time=250 ms
```

```
Reply from 6.6.6.6: bytes=56 Sequence=2 ttl=252 time=130 ms
```

```
Reply from 6.6.6.6: bytes=56 Sequence=3 ttl=252 time=110 ms
```

在 R3 上查看 IS-IS 邻居信息

```
<R3>display isis peer
```

```
[R3]dis isis peer
```

Peer information for ISIS(1)

System Id	Interface	Circuit Id	State	HoldTime	Type
R4	GE0/0/0	R3.01	Up	30s	L2(L1L2)
R1	GE0/0/1	R3.02	Up	26s	L1(L1L2)
R1	GE0/0/1	R3.02	Up	26s	L2(L1L2)
R2	GE0/0/2	R3.03	Up	25s	L1(L1L2)
R2	GE0/0/2	R3.03	Up	25s	L2(L1L2)

可以看到 R3 与 R1 既建立了 Level-1 邻接关系，又建立了 Level-2 邻接关系，R3 与 R2 既建立了 Level-1 邻接关系，又建立了 Level-2 邻接关系，但 R3 与 R4 只建立了 Level-2 邻接关系。

R1 R2 R3 是区域 10，R4 是区域 20，与 R3 区域 ID 不同，只能建立 Level-2 邻接关系。

=====

修改 IS-IS 路由器的级别

查看 LSDB 链路状态数据库<R1>display isis lsdb

Level-1 Link State Database

LSPID	Seq Num	Checksum	Holdtime	Length	ATT/P/

R1.00-00*	0x0000000a	0x810f	800	115	1/0/0
R2.00-00	0x0000000b	0xa4ca	832	115	1/0/0
R2.01-00	0x00000002	0xa2e6	832	55	0/0/0
R3.00-00	0x0000000b	0xe868	852	115	1/0/0
R3.02-00	0x00000002	0xa4e1	852	55	0/0/0
R3.03-00	0x00000002	0xb9ca	852	55	0/0/0

Level-2 Link State Database

LSPID	Seq Num	Checksum	Holdtime	Length	ATT/P/

R1.00-00*	0x0000000e	0xc2e4	800	151	0/0/0
R2.00-00	0x0000000f	0x8b08	832	151	0/0/0
R2.01-00	0x00000002	0xa2e6	832	55	0/0/0
R3.00-00	0x0000000f	0xd397	852	162	0/0/0
R3.01-00	0x00000002	0xff84	852	55	0/0/0
R3.02-00	0x00000002	0xa4e1	852	55	0/0/0
R3.03-00	0x00000002	0xb9ca	852	55	0/0/0

同时为 Level-1 和 Level-2 分别维护一个 LSDB

Level-2 的 LSDB 中不仅有 R1 所属区域的 LSP，还有其他区域的 LSP

查看 IS-IS 路由表 <R1> display isis route

1.1.1.0/24	Direct	0	0	D	1.1.1.1
1.1.1.1/32	Direct	0	0	D	127.0.0.1
2.2.2.0/24	ISIS-L1	15	10	D	192.168.12.2
6.6.6.0/24	ISIS-L2	15	40	D	192.168.13.3
127.0.0.0/8	Direct	0	0	D	127.0.0.1
127.0.0.1/32	Direct	0	0	D	127.0.0.1
192.168.12.0/24	Direct	0	0	D	192.168.12.1
192.168.12.1/32	Direct	0	0	D	127.0.0.1
192.168.13.0/24	Direct	0	0	D	192.168.13.1
192.168.13.1/32	Direct	0	0	D	127.0.0.1
192.168.23.0/24	ISIS-L1	15	20	D	192.168.12.2
	ISIS-L1	15	20	D	192.168.13.3
192.168.34.0/24	ISIS-L1	15	20	D	192.168.13.3
192.168.45.0/24	ISIS-L2	15	30	D	192.168.13.3
192.168.56.0/24	ISIS-L2	15	40	D	192.168.13.3

去往所属区域的是 Level-1 路由，非所属区域的是 Level-2 路由

R1 去往其他区域的各个网络路由是可以得到简化的，即利用缺省路由来代替，R1 完全没有必要为 Level-2 单独维护一张 LSDB 和一张 IS-IS 路由表。因此，可以在 R1 上修改为 Level-1 路由器，从而让 R1 停止为 Level-2 维护 LSDB 和路由表，实现减少系统开销并优先网络的目的。

```
R1:
isis
is-level level-1
```

配置完成后，查看 <R1> display isis lsdb，只有 Level-1 的 LSDB

Level-1 Link State Database

LSPID	Seq Num	Checksum	Holdtime	Length	ATT/P/
R1.00-00*	0x0000000b	0x8f42	1197	90	0/0/0
R2.00-00	0x0000000c	0x81ec	1196	115	1/0/0
R2.01-00	0x00000003	0xa0e7	1196	55	0/0/0
R3.00-00	0x0000000c	0xdb74	1196	115	1/0/0
R3.02-00	0x00000003	0xa2e2	1196	55	0/0/0

在 R1 上查看路由表

0.0.0.0/0	ISIS-L1	15	10	D	192.168.12.2
	ISIS-L1	15	10	D	192.168.13.3
1.1.1.0/24	Direct	0	0	D	1.1.1.1
1.1.1.1/32	Direct	0	0	D	127.0.0.1
2.2.2.0/24	ISIS-L1	15	10	D	192.168.12.2
127.0.0.0/8	Direct	0	0	D	127.0.0.1
127.0.0.1/32	Direct	0	0	D	127.0.0.1
192.168.12.0/24	Direct	0	0	D	192.168.12.1
192.168.12.1/32	Direct	0	0	D	127.0.0.1
192.168.13.0/24	Direct	0	0	D	192.168.13.1
192.168.13.1/32	Direct	0	0	D	127.0.0.1
192.168.23.0/24	ISIS-L1	15	20	D	192.168.12.2
	ISIS-L1	15	20	D	192.168.13.3
192.168.34.0/24	ISIS-L1	15	20	D	192.168.13.3

在 R2 上也进行优化

R2:

isis

is-level level-1

将 R4 配置为 Level-2 路由，R6 配置为 Level-1 路由器

```
R4:
isis
is-level level-2
```

```
R6 :
isis
is-level level-1
```

这样一来，各路由器上就不会再维护没有必要的 LSDB 和 IS-IS 路由表了

=====

修改 IS-IS 路由器接口的级别

```
terminal debugging
terminal moni
<R3>debugging isis adjacency interface g0/0/0
undo debugging all
```

```
Mar 31 2018 20:47:45.870.2-08:00 R3 ISIS/6/ISIS:
ISIS-1-ADJ: Sending L1 Hello on GE0/0/0, to SNPA 0180.c200.00
```

```
Mar 31 2018 20:47:47.10.1-08:00 R3 ISIS/6/ISIS:
ISIS-1-ADJ: Use level-2 IIH encode cache to send IIH, GE0/0/0.(I
```

```
Mar 31 2018 20:47:47.10.2-08:00 R3 ISIS/6/ISIS:
ISIS-1-ADJ: Sending L2 Hello on GE0/0/0, to SNPA 0180.c200.00
```

R3 的 g0/0/0 只需要发送 Level-2 的 Hello 消息，但现在它在发 Level-1 和 Level-2 的消息，为了让接口不再发送 Level-1 的消息，以减少链路和系统开销，可以修改接口为 Level-2

```
R3:
int g0/0/0
isis circuit-level level-2
```

配置完成后，查看一下

```
<R3>debugging isis adjacency interface g0/0/0 只发
Level-2 的 Hello 消息
```

其它的一些接口，也可以进行优化

```
R3:
int g0/0/1
isis circuit-level level-1
int g0/0/2
isis circuit-level level-1
```

```
R5:
int g0/0/0
isis circuit-level level-1
int g0/0/1
isis circuit-level level-2
```

优化之后，测试 R1 R6 的连通性，不受影响

```
<R1>ping -c 3 -a 1.1.1.1 6.6.6.6
```

=====

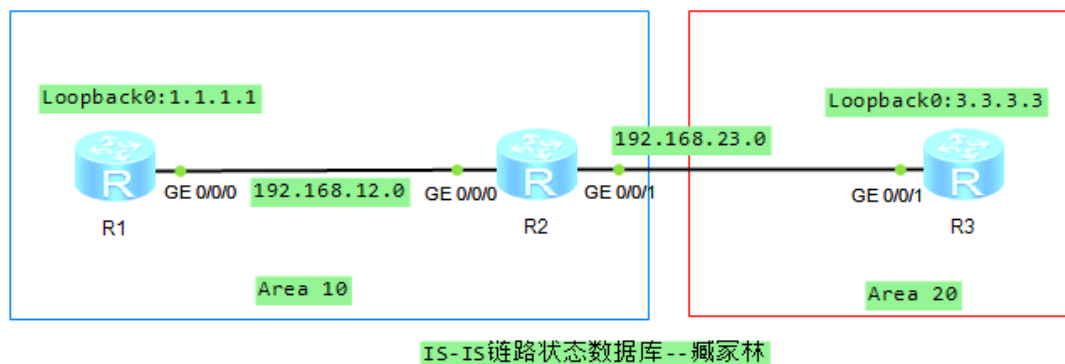
IS-IS 实验 3：IS-IS 链路状态数据库

一个 OSPF 链路状态数据库是若干条 LSA 的集合。与此相似，一个 IS-IS 链路状态数据库是若干条 LSP 的集合。与 OSPF 链路状态数据库不同，IS-IS 链路状态数据库有 level-1 和 level-2 之分。

在 IS-IS 协议中，每一条 LSP 都有一个剩余生存时间，一个序列号和一个检验和。LSP 的剩余生存时间是由最大生存时间（默认为 1200s）开始逐渐递减的。当一条 LSP 的剩余生存时间递减至 0 时，仍然会在链路状态数据库中继续保留 60s（称为 ZeroAgeLifetime），然后才会被删除。LSP 的始发路由器会周期地刷新 LSP，刷新时间间隔为 900s 减去不超过 25% 的随机量。

LSP 的序列号是一个 32 bit 的整数，初始值为 1，每次刷新时都会递增 1。与 OSPF 中的 LSA 一样，同一条 LSP，其序列号越大，表示该 LSP 越新，路由器总是将最新的 LSP 放入其链路状态数据库中。如果序列号递增至最大值时，则无法被继续刷新，但其剩余生存时间会递减至 0，然后会被从链路状态数据库中删除。

LSP 的校验和用于检验 LSP 是否在传输过程中受到损坏。当路由器收到一条包含错误的校验和的 LSP 时，会将其直接丢弃。



基本配置

R1:

```
undo ter mo
```

```
sys
sysname R1
user-interface console 0
idle-timeout 0 0
int loop 0
ip add 1.1.1.1 24
int g0/0/0
ip add 192.168.12.1 24
q
```

```
R2:
undo ter mo
sys
sysname R2
user-interface console 0
idle-timeout 0 0
int g0/0/0
ip add 192.168.12.2 24
int g0/0/1
ip add 192.168.23.2 24
q
```

```
R3:
undo ter mo
sys
sysname R3
user-interface console 0
idle-timeout 0 0
int loop 0
ip add 3.3.3.3 24
int g0/0/1
ip add 192.168.23.3 24
q
```


配置 IS-IS 协议

R1:

isis

network-entity 10.0000.0000.0001.00

is-level level-1

is-name R1

int loo 0

isis enable

int g0/0/0

isis enable

q

R2:

isis

network-entity 10.0000.0000.0002.00

is-level level-1-2

is-name R2

int g0/0/0

isis enable

isis circuit-level level-1

int g0/0/1

isis enable

isis circuit-level level-2

q

R3:

isis

network-entity 20.0000.0000.0003.00

is-level level-2

is-name R3

int loo 0

isis enable

int g0/0/1

isis enable

q

配置完成后，在 R2 上查看邻居的建立情况

[R2]display isis peer

[R2]dis isis peer

Peer information for ISIS(1)

System Id	Interface	Circuit Id	State	HoldTime	Type
R1	GE0/0/0	R2.01	Up	23s	L1
R3	GE0/0/1	R3.01	Up	8s	L2

测试 R1 R3 之间的连通性

<R1>ping -a 1.1.1.1 3.3.3.3

=====

查看 Level-1 路由器的链路状态数据库

```
[R1]dis isis lsdb
```

```
Database information for ISIS(1)
```

```
-----
```

Level-1 Link State Database

LSPID	Seq Num	Checksum	Holdtime	Length	ATT/P/OL

R1.00-00*	0x00000006	0xdee4	938	88	0/0/0
R2.00-00	0x0000000a	0x487c	998	76	1/0/0
R2.01-00	0x00000001	0xa4e5	955	55	0/0/0

第一条 LSDB

R1.00-00*,前面的 00 是伪节点标识, 00 表示此 LSP 是真实节点而非伪节点生成的, 后面的 00 为分片号, 当 LSP 的长度太长时, LSP 会被分片, 分片号的作用是为了重组被分片的 LSP。*表示些 LSP 是于本地生成的。

LSP 与 OSPF 不同, 没有 123457 类, 所有的都放在 LSP 中传递, 这也导致 LSP 会很大, 有时需要进行分片

ATT/P/OL

区域关联位 (ATT) : 用于指明一台 L2 或 L1/2 路由器具有其他区域的路由。

ATT 位的作用: 当某个区域的 L1/L2 路由器和其它区域有连接关系的时候, 它将在它产生的 L1 LSP 中设置 ATT bit, 来告诉本区域中的 L1 路由器, 它有一个区域出口点;

本区域的 L1 路由器利用 L1 LSPDB 中的 ATT bit 来作为可能的区域出口点；如果有多个，选择一个离自己最近的设置了 ATT bit 的 L1/L2 路由器作为区域的缺省出口点，对于 IP 路由来说，具体表现就是 L1 路由安装一条指向该 L1/L2 路由的缺省路由 0.0.0.0/0 在路由表中。

区域修复 (Partition Repair) 指 L1 区域被分裂后的修复，如果该位设置为 1，表示源路由支持区域修复。

通过骨干区域创建虚连接实现 L1 区域的修复，虚连接的端点是在该 L1 区域的两台 L1/2 路由器上。华为目前的 VRP 系统不支持区域修复功能，我们看到的 LSP 中的 P 位为 0

OverLoad(过载)：OverLoad 位表示始发路由器的数据库是否过载。

一般情况下为 0，如果设置为 1，则表示 LSP 始发源路由器的 CPU 和内存资源已经不足，无法维护一个完整的链路状态数据库，数据流经过这台设备时有可能得不到正常的转发。

R2:

isis

set-overload

```
[R1]dis isis lsdb
```

Database information for ISIS(1)

Level-1 Link State Database

LSPID	Seq Num	Checksum	Holdtime	Length	ATT/P/OL

R1.00-00*	0x00000006	0xdee4	824	88	0/0/0
R2.00-00	0x0000000b	0x4285	1196	76	0/0/1
R2.01-00	0x00000001	0xa4e5	841	55	0/0/0

查看详细信息 <R1>display isis lsdb verbose

```
[R1]dis isis lsdb verbose
```

Database information for ISIS(1)

Level-1 Link State Database

LSPID	Seq Num	Checksum	Holdtime	Length

0000.0000.0001.00-00*	0x00000006	0xdee4	780	88
SOURCE	R1.00			
HOST NAME	R1			
NLPID	IPV4			
AREA ADDR	10			
INTF ADDR	1.1.1.1			
INTF ADDR	192.168.12.1			

=====

在 R2 上查看 Level-2 路由器的链路状态数据库

```
<R2>display isis lsdb
```

Level-1 Link State Database

LSPID	Seq Num	Checksum	Holdtime	Length
R1.00-00	0x00000006	0xdee4	714	88
R2.00-00*	0x0000000b	0x4285	1086	76
R2.01-00*	0x00000001	0xa4e5	732	55

Total LSP(s): 3

*(In TLV)-Leaking Route, *(By LSPID)-Self LSP, +-Self LSP(Extended)
ATT-Attached, P-Partition, OL-Overload

Level-2 Link State Database

LSPID	Seq Num	Checksum	Holdtime	Length
R2.00-00*	0x00000011	0x5b4d	1086	88
R3.00-00	0x00000009	0xcab8	774	88
R3.01-00	0x00000001	0xc9bd	774	55

分别维护 Level-1 和 Level-2 链路状态数据库，Level-1 的与 R1 完全相同（除了*），这表明 Level-1 链路状态数据库在 R1 和 R2 上完成了同步

<R2>display isis lsdb level-1 verbose

<R2>display isis lsdb level-2 verbose

Level-1 和 Level-2 的主要区别在于：Level-1 中 LSDB 区域 ID 彼此相同，而 Level-2 区域 ID 可以不同

```

0000.0000.0002.00-00* 0x00000011  0x5b4d      1020
SOURCE          R2.00
HOST NAME       R2
NLPID           IPV4
AREA ADDR       10
INTF ADDR       192.168.12.2
INTF ADDR       192.168.23.2
NBR ID          R3.01          COST: 10
IP-Internal     192.168.12.0    255.255.255.0    COST: 10
IP-Internal     192.168.23.0    255.255.255.0    COST: 10

0000.0000.0003.00-00 0x00000009  0xcab8      708
SOURCE          R3.00
HOST NAME       R3
NLPID           IPV4
AREA ADDR       20
INTF ADDR       192.168.23.3

```

= = = = =

IS-IS 实验 4：IS-IS DIS

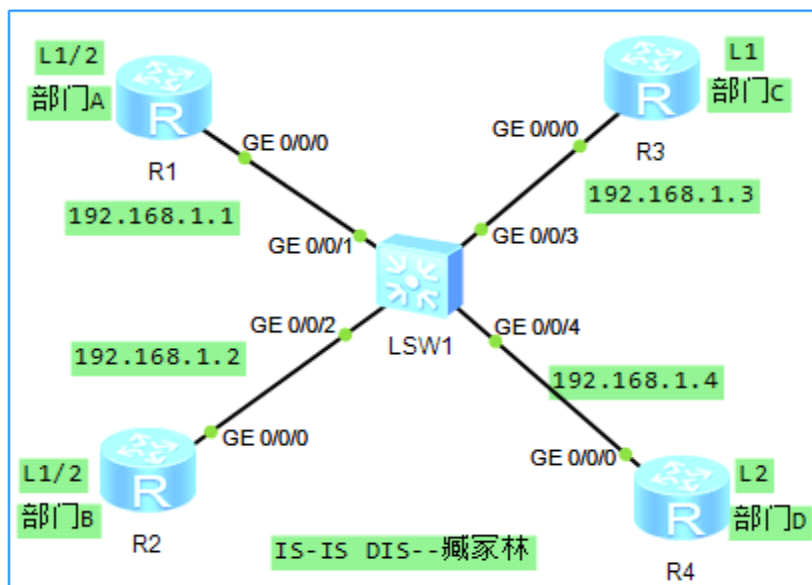
IS-IS 协议只支持两种网络类型，即广播网络和点到点网络。与 OSPF 协议相同，IS-IS 协议在广播网络中会将网络视为一个伪节点（Pseudonode，简称 PSN），并选举出一台 DIS（Designated IS）路由器来代行这个伪节点的职责。

DIS 的作用与 OSPF 的 DR 类似，可以减少不必要的 LSP 泛洪，DIS 选举是抢占性的。选举的依据是比较同一网络中路由器接口的 DIS 优先级，其次是比较接口的 MAC 地址。路由器

的 IS-IS 接口都拥有一个 level-1 DIS 优先级和一个 level-2 DIS 优先级，默认是 64，取值范围是 0--127。如果 DIS 优先级的值为 0，并不表示不参与 DIS 选举，而只是表示 DIS 优先级最低。在接口的 DIS 优先级相同的情况下，MAC 地址较大者将成为 DIS。

选举 DIS 后，广播网络中的路由器仍然需要与所有的邻居建立邻接关系，而不仅仅是与 DIS 建立邻接关系。在广播网络中，DIS 会周期性（默认为 10s）地发送携带 CSNP 消息的组播帧来实现链路状态数据库之间的同步，其中 level-1 DIS 使用的组播 MAC 地址为 018.C200.0014，而 level-2 DIS 使用的组播 MAC 地址为 018.C200.0015

在 IS-IS 协议中，只有 DIS，没有 Backup DIS



实验中的 MAC 地址是随机生成的
基本配置

```
R1:  
undo ter mo  
sys  
sysname R1  
user-interface console 0  
idle-timeout 0 0  
int g0/0/0  
ip add 192.168.1.1 24  
q
```

```
isis  
network-entity 10.0000.0000.0001.00  
is-name R1  
int g0/0/0  
isis enable  
q
```

```
R2:  
undo ter mo  
sys  
sysname R2  
user-interface console 0  
idle-timeout 0 0  
int g0/0/0  
ip add 192.168.1.2 24  
q
```

```
isis  
network-entity 10.0000.0000.0002.00  
is-name R2  
int g0/0/0  
isis enable  
q
```

R3:

undo ter mo

sys

sysname R3

user-interface console 0

idle-timeout 0 0

int g0/0/0

ip add 192.168.1.3 24

q

isis

network-entity 10.0000.0000.0003.00

is-name R3

is-level level-1

int g0/0/0

isis enable

q

R4:

undo ter mo

sys

sysname R4

user-interface console 0

idle-timeout 0 0

int g0/0/0

ip add 192.168.1.4 24

q

isis

network-entity 10.0000.0000.0004.00

is-name R4

is-level level-2

int g0/0/0

isis enable

q

在 R1 上查看 IS-IS 邻居信息

<R1>display isis peer

Peer information for ISIS(1)

System Id	Interface	Circuit Id	State	HoldTime	Type	PRI

R2	GE0/0/0	R3.01	Up	26s	L1(L1L2)	64
R3	GE0/0/0	R3.01	Up	20s	L1	64
R2	GE0/0/0	R4.01	Up	22s	L2(L1L2)	64
R4	GE0/0/0	R4.01	Up	30s	L2	64

邻居关系建立正常

=====

查看默认选举的 DIS

<R1>display isis interface 1

[R1]dis isis interface

Interface information for ISIS(1)

Interface	Id	IPV4.State	IPV6.State	MTU	Type	DIS

GE0/0/0	001	Up	Down	1497	L1/L2	No/No

```
[R2]dis isis int
```

Interface information for ISIS(1)

Interface	Id	IPV4.State	IPV6.State	MTU	Type	DIS
GE0/0/0	001	Up	Down	1497	L1/L2	No/No

```
[R3]dis isis interface
```

Interface information for ISIS(1)

Interface	Id	IPV4.State	IPV6.State	MTU	Type	DIS
GE0/0/0	001	Up	Down	1497	L1/L2	Yes/No

```
[R4]dis isis interface
```

Interface information for ISIS(1)

Interface	Id	IPV4.State	IPV6.State	MTU	Type	DIS
GE0/0/0	001	Up	Down	1497	L1/L2	No/Yes

从上面可以看，R3 被选举为 Level-1 的 DRS，R4 被选举为 Level-2 的 DIS

```
<R1>display isis interface verbose
```

```
[R1]dis isis interface verbose
```

```
Interface information for ISIS(1)
-----
Interface      Id      IPV4.State      IPV6.State      MTU  Type  DIS
GE0/0/0        001      Up              Down            1497 L1/L2 No/No
Circuit MT State      : Standard
Description           :
SNPA Address          : 5489-9810-5a6e
IP Address             : 192.168.1.1
```

伪节点只是一个逻辑上的概念，用来表示一个广播型网络本身，而 DIS 路由器是连接到这个广播网络的一台路由器，DIS 和伪节点是两个不同的概念，只是 DIS 代行了伪节点的职责而已。

优先级相同，mac 地址大的，成为 DIS 可能性大一些

=====

修改 DIS 优先级来控制 DIS 选举结果

让 R2 成为 Level-2 的 DIS，在接口下修改优先级

R2:

```
int g0/0/0
```

```
isis dis-priority 127 level-2
```

配置完成后，来查看一下

```
<R2>display isis interface verbose
```

DIS 优先级修改之后，优先级高的路由器会迅速抢占 DIS 角色
默认是抢占的

Interface	Id	IPv4.State	IPv6.State	MTU	Type	DIS
GE0/0/0	001	Up	Down	1497	L1/L2	No/Yes
Circuit MT State		: Standard				
Description		:				
SNPA Address		: 5489-9859-7f39				
IP Address		: 192.168.1.2				
IPv6 Link Local Address		:				
IPv6 Global Address(es)		:				
Csnr Timer Value		: L1	10	L2	10	
Hello Timer Value		: L1	10	L2	10	
DIS Hello Timer Value		: L1	3	L2	3	
Hello Multiplier Value		: L1	3	L2	3	
LSP-Throttle Timer		: L12	50			
Cost		: L1	10	L2	10	
Ipv6 Cost		: L1	10	L2	10	
Priority		: L1	64	L2	127	

= = = = =

IS-IS 实验 5 : IS-IS 开销值和协议优先级

IS-IS 协议为路由器的每个 IS-IS 接口定义并维护了一个 level-1 开销值和一个 level-2 开销值。开销值可以在接口上或者全局上手动配置，也可以使用 auto-cost 自动计算确定。

```
int g0/0/0
isis cost 50
```

也可以修改全局的开销值

```
isis
circuit-cost 30
```

isis

bandwidth-reference 1000

IS-IS 自动计算链路度量值时依据的带宽参考值为 100Mbps。

isis

auto-cost enable

IS-IS 有三种方式来确定接口的开销，按照优先级由高到底分别如下：

- ① 接口开销：为单个接口设置开销，优先级最高。
- ② 全局开销：为所有接口设置开销，优先级中等。
- ③ 自动计算开销：根据接口带宽自动计算开销，优先级最低。

用户可根据需要选择一种或多种接口开销配置方式。在配置接口开销前，可根据实际需要配置 IS-IS 的开销类型，因为不同类型的开销的取值范围不一样。

如果没有为 IS-IS 接口配置任何开销值，IS-IS 接口的默认开销为 10，开销类型为 narrow。在实际应用中，为方便 IS-IS 实现其扩展功能，通常将 IS-IS 的路由开销类型设置为 wide 模式。

采用 auto-cost 计算确定接口开销值时，如果开销值类型为 wide，则接口开销值 = (参考带宽 / 接口带宽) x 10，如果开销值类型为 narrow，则接口开销值为与接口带宽绑定的固定值。开销值为 narrow 时，接口带宽分为几个档次，依次为小于等于 10 MB，大于 10 MB 小于等于 100 MB，等 6 种，而相应的接口开销值为 60, 50 等 6 种。在没有任何配置的情况下，IS-IS 开销类型默认为 narrow，且所有带宽档次的接口默认开

销值均为 10 。

IS-IS 开销类型 narrow ，接口下最大只能配置值为 63 的开销值，但是在大型网络设计中，较小的度量范围不能满足实际需求。IS-IS 开销类型 wide ，接口开销值可以扩展到 16777215 。

isis

cost-style narrow

auto-cost enable

自动计算开销，g0/0/0 接口开销为 20

isis

cost-style wide

auto-cost enable

自动计算开销，g0/0/0 接口开销为 1

配置 IS-IS 接口开销类型

①narrow：指定 IS-IS 设备所有接口只能接收和发送开销类型为 narrow 的路由。

Narrow 模式下路由的开销值取值范围为 1~63 的整数。

②wide：指定 IS-IS 设备所有接口只能接收和发送开销类型为 wide 的路由。

wide 模式下路由的开销值取值范围为 1~16777215 的整数。

③wide-compatible：指定 IS-IS 设备所有接口可以接收开销类型为 narrow 和 wide 的路由，

但却只发送开销类型为 wide 的路由。

④narrow-compatible：指定 IS-IS 设备所有接口可以接收开销类型为 narrow 和 wide 的路由，

但却只发送开销类型为 narrow 的路由。

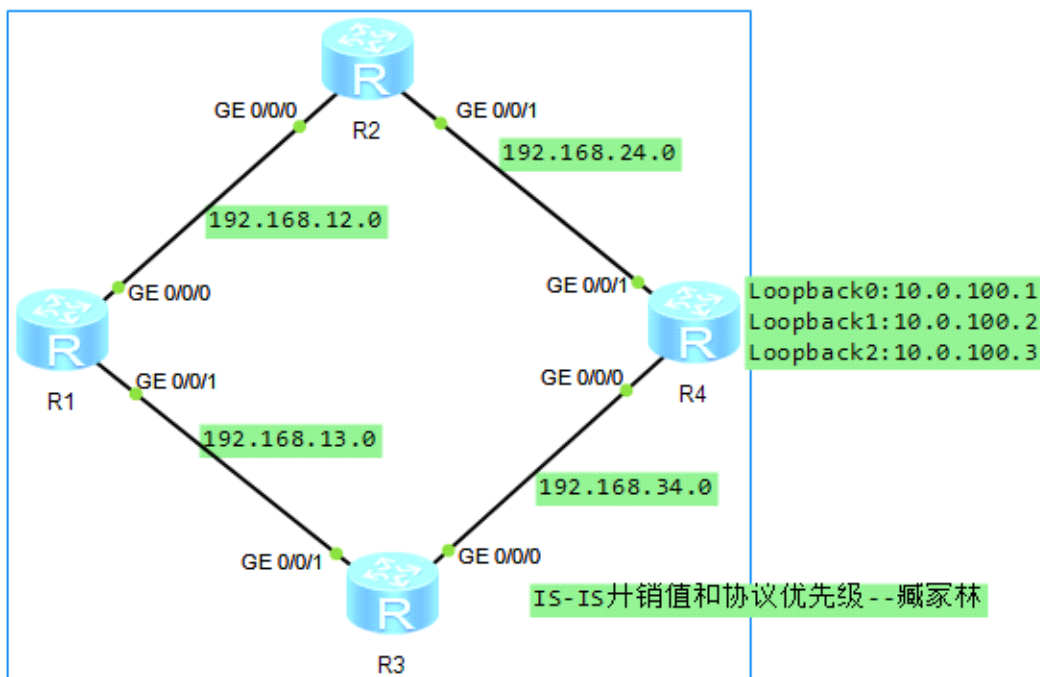
⑤compatible：指定 IS-IS 设备所有接口可以接收和发送开销类型为 narrow 和 wide 的路由。

isis

cost-style wide

wide 类型下的 IS-IS 和 narrow 类型下的 IS-IS 不可实现互通。如果需要互通，就必须设置成一致的开销类型，让网络上所有路由器都可以接收其他路由器发的所有报文。

IS-IS 路由协议的优先级值为 15，IS-IS 也拥有一系统的计时器，其中 hello timer 是用来控制 IS-IS hello 报文发送的时间间隔的。



都在同一个区域，路由器设置为 level-1

修改接口开销，接口类型，网络类型，协议优先级，hello 时间

基本配置

```
R1:
undo ter mo
sys
sysname R1
user-interface console 0
idle-timeout 0 0
int g0/0/0
ip add 192.168.12.1 24
int g0/0/1
ip add 192.168.13.1 24
q
```

R2:
undo ter mo
sys
sysname R2
user-interface console 0
idle-timeout 0 0
int g0/0/0
ip add 192.168.12.2 24
int g0/0/1
ip add 192.168.24.2 24
q

R3:
undo ter mo
sys
sysname R3
user-interface console 0
idle-timeout 0 0
int g0/0/1
ip add 192.168.13.3 24
int g0/0/0
ip add 192.168.34.3 24
q

R4:
undo ter mo
sys
sysname R4
user-interface console 0
idle-timeout 0 0
int loop 0
ip add 10.0.100.1 32
int loop 1
ip add 10.0.100.2 32

```
int loop 2
ip add 10.0.100.3 32
int g0/0/1
ip add 192.168.24.4 24
int g0/0/0
ip add 192.168.34.4 24
q
```

配置 IS-IS 协议

R1:

```
isis
network-entity 10.0000.0000.0001.00
is-name R1
is-level level-1
int g0/0/0
isis enable
int g0/0/1
isis enable
```

R2:

```
isis
network-entity 10.0000.0000.0002.00
is-name R2
is-level level-1
int g0/0/0
isis enable
int g0/0/1
isis enable
```

R3:

```
isis
network-entity 10.0000.0000.0003.00
is-name R3
is-level level-1
```

```
int g0/0/0
isis enable
int g0/0/1
isis enable
```

```
R4:
isis
network-entity 10.0000.0000.0004.00
is-name R4
is-level level-1
int loo 0
isis enable
int loo 1
isis enable
int loo 2
isis enable
int g0/0/0
isis enable
int g0/0/1
isis enable
```

配置完成后，查看一下邻居

=====

修改开销值

<R1>display isis route

IPv4 Destination	IntCost	ExtCost	ExitInterface	NextHop
10.0.100.2/32	20	NULL	GE0/0/0	192.168.12.2
			GE0/0/1	192.168.13.3
10.0.100.1/32	20	NULL	GE0/0/0	192.168.12.2
			GE0/0/1	192.168.13.3
192.168.34.0/24	20	NULL	GE0/0/1	192.168.13.3
192.168.24.0/24	20	NULL	GE0/0/0	192.168.12.2
192.168.13.0/24	10	NULL	GE0/0/1	Direct
10.0.100.3/32	20	NULL	GE0/0/0	192.168.12.2
			GE0/0/1	192.168.13.3
192.168.12.0/24	10	NULL	GE0/0/0	Direct

在 R1 上看到 R4 的环回口是负载均衡的。

<R1>display isis interface verbose

```

IPV6 Link Local Address      :
IPV6 Global Address(es)     :
Csnp Timer Value             : L1      10  L2      10
Hello Timer Value            : L1      10  L2      10
DIS Hello Timer Value        : L1       3  L2       3
Hello Multiplier Value       : L1       3  L2       3
LSP-Throttle Timer           : L12     50
Cost                          : L1      10  L2      10
Ipv6 Cost                    : L1      10  L2      10
Priority                      : L1     64  L2     64

```

开销值默认为 10，修改 cost 值，让数据从 R3 转发
把到 R2 的 g0/0/0 cost 值改大一些

R1：

```
int g0/0/0
isis cost 50 level-1
```

IPv4 Destination	IntCost	ExtCost	ExitInterface	NextHop
10.0.100.2/32	20	NULL	GE0/0/1	192.168.13.3
10.0.100.1/32	20	NULL	GE0/0/1	192.168.13.3
192.168.34.0/24	20	NULL	GE0/0/1	192.168.13.3
192.168.24.0/24	30	NULL	GE0/0/1	192.168.13.3
192.168.13.0/24	10	NULL	GE0/0/1	Direct
10.0.100.3/32	20	NULL	GE0/0/1	192.168.13.3

查看效果，现在都是从 R3 走

```
<R1>tracert 10.0.100.1
```

```
<R1>tracert 10.0.100.1
```

```
tracert to 10.0.100.1(10.0.100.1), max hops:
RL_C to break
```

```
1 192.168.13.3 50 ms 40 ms 50 ms
```

```
2 192.168.34.4 90 ms 70 ms 60 ms
```

```
<R1>
```

```
=====
```

修改 IS-IS 接口开销类型

```
isis
```

```
cost-style wide
```



```
int g0/0/0
isis cost <1-16777215>
```

两端类型不一致，邻居可以建立，但路由不能学习

narrow 类型下的 LSDB
dis isis lsdb verbose

INTF ADDR	10.0.100.3		
INTF ADDR	192.168.34.4		
INTF ADDR	192.168.24.4		
NBR ID	R4.02	COST: 10	
NBR ID	R4.01	COST: 10	
IP-Internal	10.0.100.1	255.255.255.255	COST: 0
IP-Internal	10.0.100.2	255.255.255.255	COST: 0
IP-Internal	10.0.100.3	255.255.255.255	COST: 0
IP-Internal	192.168.34.0	255.255.255.0	COST: 10
IP-Internal	192.168.24.0	255.255.255.0	COST: 10

wide 类型下的 LSDB

```

-----
0000.0000.0001.00-00* 0x0000000c 0x3a6d 1194
SOURCE R1.00
HOST NAME R1
NLPID IPV4
AREA ADDR 10
INTF ADDR 192.168.12.1
INTF ADDR 192.168.13.1
+NBR ID R1.02 COST: 10
+NBR ID R1.01 COST: 50
+IP-Extended 192.168.12.0 255.255.255.0 COST: 50
+IP-Extended 192.168.13.0 255.255.255.0 COST: 10

```

=====

修改网络类型

```

int g0/0/0
isis circuit-type p2p

```

配置命令中，只有 p2p，也就是说只能将广播类型，改为 p2p
两端网络类型一致，才能建立 ISIS 邻居

=====

修改协议优先级

在 R1 上添加静态路由，从 R2 去 R4 的 10.0.100.1

R1:

```

ip route-static 10.0.100.1 32 192.168.12.2

```

[R1]display ip routing-table 还是通过 IS-IS 学习的路由
静态路由优先级 60，IS-IS 为 15，越小越优

<R1>display default-parameter isis

Cost-Style	: Narrow
Circuit-Cost <IPv4>	: L1 10 L2 10
Circuit-Cost <IPv6>	: L1 10 L2 10
IS-Level	: L12
LSP-Originate-Length	: 1497
LSP-Receive-Length	: 1497
LSP-Max-Age <s>	: 1200
LSP-Generation-IntelliTimer <s,ms,ms>	: L1 Max 2 Init 0 Incr 0 L2 Max 2 Init 0 Incr 0
LSP-Refresh-Interval <s>	: 900
Preference	: IPv4 15 IPv6 15
SPF-IntelliTimer <s,ms,ms>	: Max 5 Init 50 Incr 200

修改为 70

R1 :

isis

preference 70

10.0.100.0/24	Static	60	0	RD	192.168.12.2
10.0.100.1/32	ISIS-L1	70	20	D	192.168.13.3
10.0.100.2/32	ISIS-L1	70	20	D	192.168.13.3
10.0.100.3/32	ISIS-L1	70	20	D	192.168.13.3
127.0.0.0/8	Direct	0	0	D	127.0.0.1
127.0.0.1/32	Direct	0	0	D	127.0.0.1
192.168.12.0/24	Direct	0	0	D	192.168.12.1
192.168.12.1/32	Direct	0	0	D	127.0.0.1
192.168.13.0/24	Direct	0	0	D	192.168.13.1
192.168.13.1/32	Direct	0	0	D	127.0.0.1
192.168.24.0/24	ISIS-L1	70	30	D	192.168.13.3
192.168.34.0/24	ISIS-L1	70	20	D	192.168.13.3

现在 R1 去往 10.0.100.1 从 R2 走
 去往 10.0.100.2 和 10.0.100.3 从 R3 走

=====

修改 Hello Timer 设定值

为了减少在链路上发送 IS-IS hello 报文的频率，可以人为地增大报文的时间间隔。默认 hello 为 10s, DIS 为 3s(自动取相应值的三分之一，并取整)

<R1>display isis interface verbose

IPV6 Global Address(es)	:				
Csnp Timer Value	:	L1	10	L2	10
Hello Timer Value	:	L1	10	L2	10
DIS Hello Timer Value	:	L1	3	L2	3
Hello Multiplier Value	:	L1	3	L2	3
LSP-Throttle Timer	:	L12	50		
Cost	:	L1	10	L2	10

```
terminal debugging
debugging isis adjacency interface g0/0/0
查看一下 debug 的情况
```

修改为 30s

R1:

```
int g0/0/0
```

```
isis timer hello 30 level-1
```

IPV6 Global Address(es)	:				
Csnp Timer Value	:	L1	10	L2	10
Hello Timer Value	:	L1	30	L2	30
DIS Hello Timer Value	:	L1	10	L2	10
Hello Multiplier Value	:	L1	3	L2	3
LSP-Throttle Timer	:	L12	50		

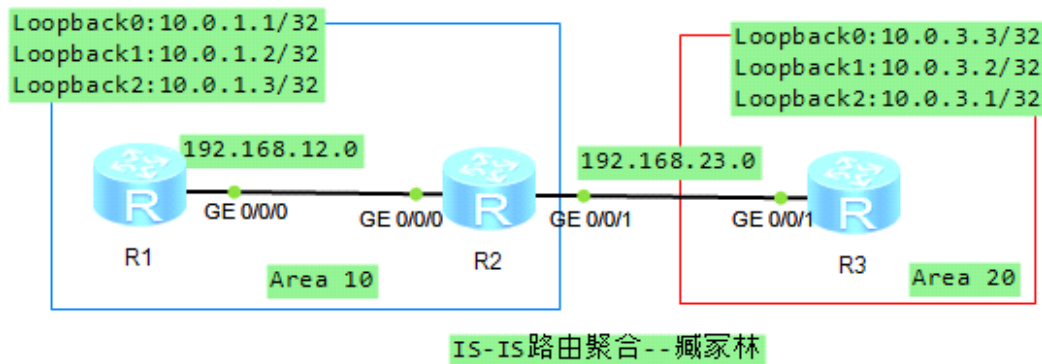
= = = = =

IS-IS 实验 6：IS-IS 路由汇总

与 OSPF 协议相同，IS-IS 也能够通过路由聚合来减少路由条目。不同的是，OSPF 只能在 ABR 和 ASBR 路由器上进行路由聚合，而 IS-IS 路由器能否进行路由聚合以及对什么样的路由才能进行聚合取决于路由器的类型及路由的类型。

在 IS-IS 网络中，level-1 路由器只维护 level-1 链路状态数据库，只能对相应的 level-1 的直连路由进行聚合，并将聚合后的路由以 level-1 LSP 的形式发送给其他路由。level-2 路由器只维护 level-2 链路状态数据库，只能对相应的 level-2 的直连路由进行聚合，并将聚合后的路由以 level-2 LSP 的形式发送给其他路由。

level-1-2 路由器分别维护 level-1 和 level-2 链路状态数据库，level-1-2 路由器能将 level-1 路由表中的路由（不必一定是直连路由）进行聚合后以 level-1 LSP 的形式发送给其他路由器，将 level-2 路由表中的路由（不必一定是直连路由）进行聚合后以 level-2 LSP 的形式发送给其他路由器，并且还能够将 level-1 路由表中的路由（不必一定是直连路由）聚合以后以 level-2 LSP 的形式发送给其他路由器。



基本配置

R1:

```
undo ter mo
sys
sysname R1
user-interface console 0
idle-timeout 0 0
int loop 0
ip add 10.0.1.1 32
int loop 1
ip add 10.0.1.2 32
int loop 2
ip add 10.0.1.3 32
int g0/0/0
ip add 192.168.12.1 24
q
```

```
R2:
undo ter mo
sys
sysname R2
user-interface console 0
idle-timeout 0 0
int g0/0/0
ip add 192.168.12.2 24
int g0/0/1
ip add 192.168.23.2 24
q
```

```
R3:
undo ter mo
sys
sysname R3
user-interface console 0
idle-timeout 0 0
int loop 0
ip add 10.0.3.3 32
int loop 1
ip add 10.0.3.2 32
int loop 2
ip add 10.0.3.1 32
int g0/0/1
ip add 192.168.23.3 24
q
```

配置 IS-IS 协议

```
R1:
isis
network-entity 10.0000.0000.0001.00
is-name R1
```

```
is-level level-1
int loo 0
isis enable
int loo 1
isis enable
int loo 2
isis enable
int g0/0/0
isis enable
```

```
R2:
isis
network-entity 10.0000.0000.0002.00
is-name R2
int g0/0/0
isis enable
isis circuit-level level-1
int g0/0/1
isis enable
isis circuit-level level-2
```

```
R3:
isis
network-entity 20.0000.0000.0003.00
is-name R3
is-level level-2
int loo 0
isis enable
int loo 1
isis enable
int loo 2
isis enable
int g0/0/1
```


isis enable

q

查看邻居的建立

=====

在 Level-1 路由器上进行路由聚合

<R2>display isis route

IPV4 Destination	IntCost	ExtCost	ExitInterface	NextHop	Flags

10.0.1.3/32	10	NULL	GE0/0/0	192.168.12.1	A/-/L/-
10.0.1.2/32	10	NULL	GE0/0/0	192.168.12.1	A/-/L/-
10.0.1.1/32	10	NULL	GE0/0/0	192.168.12.1	A/-/L/-
192.168.12.0/24	10	NULL	GE0/0/0	Direct	D/-/L/-

Flags”为路由信息标记，不同路由标记具体如下：

①D：表示直连路由

②A：表示此路由被加入单播路由表中。

③L：表示此路由通过 LSP 发布出去。

R1：

isis

summary 10.0.1.0 255.255.255.252

路由表没有任何变化

默认情况下，IS-IS 总是将聚合后的路由以 level-2 LSP 的形式传递给其它路由器，而 R1 是一个 level-1 路由器，无法以 level-2 LSP 形式传递，所以没有变化。

R1 :
isis
summary 10.0.1.0 255.255.255.252 level-1

在 R2 查看 IS-IS 路由表

IPv4 Destination	IntCost	ExtCost	ExitInterface	NextHop

192.168.23.0/24	10	NULL	GE0/0/1	Direct
10.0.1.0/30	10	NULL	GE0/0/0	192.168.12.1
192.168.12.0/24	10	NULL	GE0/0/0	Direct

在 R3 上也只有 10.0.1.0 的聚合

=====

在 Level-2 路由器上进行路由聚合

R3 :
isis
summary 10.0.3.0 255.255.255.252 level-2

在 R2 上进行检查，只有聚合后的路由

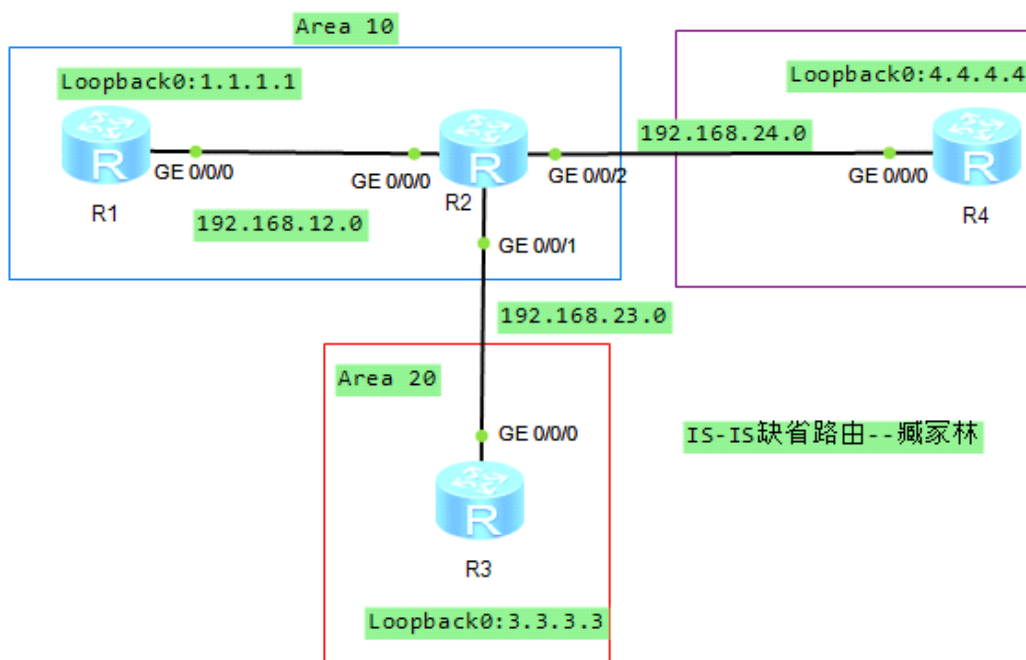
IPv4 Destination	IntCost	ExtCost	ExitInterface	NextHop

10.0.3.0/30	10	NULL	GE0/0/1	192.168.23.3
192.168.23.0/24	10	NULL	GE0/0/1	Direct
192.168.12.0/24	10	NULL	GE0/0/0	Direct

=====

IS-IS 实验 7：IS-IS 缺省路由

IS-IS 有两种缺省路由，第一种缺省路由是由 level-1 路由器在特定条件下自动产生的，它的下一跳是离它最近的（cost 最小）level-1-2 路由器。第二种缺省路由是 IS-IS 路由器上使用 default-route-advertise 命令产生并发布的。



R4 不运行 IS-IS 协议，R2 写一条默认路由到 R4

基本配置

```
R1:  
undo ter mo  
sys  
sysname R1  
user-interface console 0  
idle-timeout 0 0
```

```
int loop 0
ip add 1.1.1.1 24
int g0/0/0
ip add 192.168.12.1 24
q
```

```
R2:
undo ter mo
sys
sysname R2
user-interface console 0
idle-timeout 0 0
int g0/0/0
ip add 192.168.12.2 24
int g0/0/1
ip add 192.168.23.2 24
int g0/0/2
ip add 192.168.24.2 24
q
```

```
R3:
undo ter mo
sys
sysname R3
user-interface console 0
idle-timeout 0 0
int loop 0
ip add 3.3.3.3 24
int g0/0/0
ip add 192.168.23.3 24
q
```

```
R4:
undo ter mo
```

```
sys
sysname R4
user-interface console 0
idle-timeout 0 0
int loop 0
ip add 4.4.4.4 24
int g0/0/0
ip add 192.168.24.4 24
q
```

配置 IS-IS 协议

```
R1:
isis
network-entity 10.0000.0000.0001.00
is-name R1
is-level level-1
int loo 0
isis enable
int g0/0/0
isis enable
q
```

```
R2:
isis
network-entity 10.0000.0000.0002.00
is-name R2
int g0/0/0
isis enable
isis circuit-level level-1
int g0/0/1
isis enable
isis circuit-level level-2
q
```

```
R3:
isis
network-entity 20.0000.0000.0003.00
is-name R3
is-level level-2
int loo 0
isis enable
int g0/0/0
isis enable
q
```

R2 查看邻居的建立，查看路由表

=====

在 R2 R4 上配置静态路由

```
R2 :
ip route-static 0.0.0.0 0 192.168.24.4
```

```
R4 :
ip route-static 1.1.1.0 24 192.168.24.2
ip route-static 3.3.3.0 24 192.168.24.2
```

测试 R1 R4 的连通性<R1>ping -a 1.1.1.1 4.4.4.4

```
[R1]ping -a 1.1.1.1 4.4.4.4
  PING 4.4.4.4: 56 data bytes, press CTRL_C to break
    Reply from 4.4.4.4: bytes=56 Sequence=1 ttl=254 time=130 ms
    Reply from 4.4.4.4: bytes=56 Sequence=2 ttl=254 time=80 ms
    Reply from 4.4.4.4: bytes=56 Sequence=3 ttl=254 time=50 ms
    Reply from 4.4.4.4: bytes=56 Sequence=4 ttl=254 time=60 ms
```

R1 上没有明细路由，只有缺省路由

0.0.0.0/0	ISIS-L1	15	10	D	192.168.12.2
1.1.1.0/24	Direct	0	0	D	1.1.1.1
1.1.1.1/32	Direct	0	0	D	127.0.0.1
127.0.0.0/8	Direct	0	0	D	127.0.0.1
127.0.0.1/32	Direct	0	0	D	127.0.0.1
192.168.12.0/24	Direct	0	0	D	192.168.12.1
192.168.12.1/32	Direct	0	0	D	127.0.0.1
192.168.23.0/24	ISIS-L1	15	20	D	192.168.12.2

<R1>display isis lsdb

LSPID	Seq Num	Checksum	Holdtime	Length	ATT/P/
R1.00-00*	0x00000005	0xc7fd	1070	88	0/0/0
R1.01-00*	0x00000001	0xb5d7	1070	55	0/0/0
R2.00-00	0x00000006	0x436e	1085	88	1/0/0

R2 产生的，ATT 位被设置为 1 的 Level-1 LSP

L1/L2 router 发送到本 area 的 LSP 中若将 att 置位，则表示该 L1/L2 可以通往外部

但如果 R2 连接骨干区域的链路发生故障，R2 将不会再把自
己产生的 Level-1 LSP 的 ATT 位设置为 1，而设置为 0

R2 :
int g0/0/1
shutdown

<R1>display isis lsdb

LSPID	Seq Num	Checksum	Holdtime	Length	ATT/P/
R1.00-00*	0x00000005	0xc7fd	960	88	0/0/0
R1.01-00*	0x00000001	0xb5d7	960	55	0/0/0
R2.00-00	0x00000008	0xd481	1193	72	0/0/0

现在也没有默认路由

Destination/Mask	Proto	Pre	Cost	Flags	NextHop
1.1.1.0/24	Direct	0	0	D	1.1.1.1
1.1.1.1/32	Direct	0	0	D	127.0.0.1
127.0.0.0/8	Direct	0	0	D	127.0.0.1
127.0.0.1/32	Direct	0	0	D	127.0.0.1

=====

手动向 R1 发布缺省路由

现在 R1 与 R4 之间是不通的。为了解决这个问题，可以在 R2 上手动强制向 R1 发布缺省路由。默认只向 Level-2 发布，需要加上 Level-1 关键字

```
R2 :
isis
default-route-advertise level-1
```

如果 R2 上没有默认路由，也可以加上 always，变成 default-route-advertise always level-1
always 与 match default 只能使用一个

<R1>display isis route 有默认路由

关闭 R2 的 g0/0/2 接口

R2 :

```
int g0/0/2
```

```
shutdown
```

在 R1 上依然有缺省路由。

默认情况下，路由器使用 default-route-advertise 命令发布缺省路由，无论自己是否有，都向邻接关系路由器发布。

但这样是不正确的，需要改进，将自己有缺省路由时，才发布

R2 :

```
isis
```

```
default-route-advertise match default level-1
```

配置之后，R1 不再有缺省路由。

把 R2 R4 之间链路打开

R2 :

```
int g0/0/2
```

```
undo shutdown
```

R1 上有缺省路由了

=====

手动向 R3 发布缺省路由

打开 R2 的 g0/0/1

R2 :

```
int g0/0/1
```

```
undo shutdown
```

同时向 R1 R3 发布缺省路由

R2 :

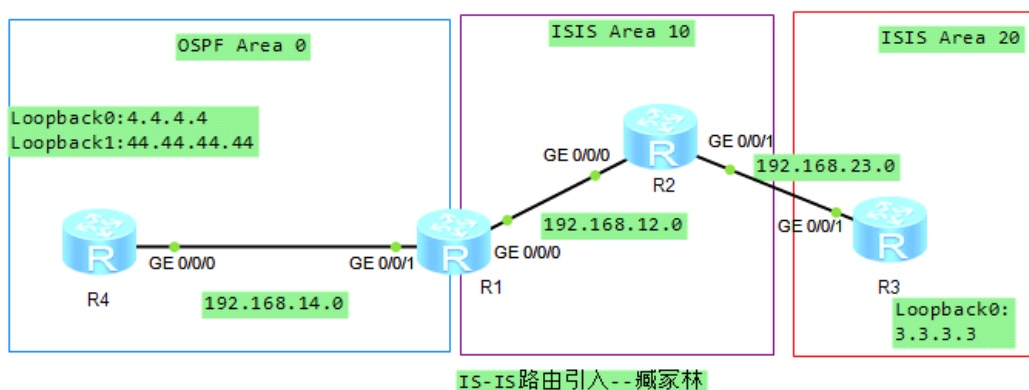
isis

default-route-advertise match default level-1-2

= = = = =

IS-IS 实验 8 : IS-IS 路由引入

IS-IS 网络能够引入其他路由协议的路由和其他 IS-IS 协议进程的路由。默认情况下，IS-IS 总是以 level-2 路由类型引入外部路由。但是，通过手动配置，也可以以 level-1 路由类型引入外部路由。IS-IS 协议在引入外部路由时，可以手动配置引入路由的开销值，并可以使用 route-policy 对引入的路由进行过滤。



基本配置

R1:

undo ter mo

sys

sysname R1

```
user-interface console 0
idle-timeout 0 0
int g0/0/1
ip add 192.168.14.1 24
int g0/0/0
ip add 192.168.12.1 24
q
```

```
R2:
undo ter mo
sys
sysname R2
user-interface console 0
idle-timeout 0 0
int g0/0/0
ip add 192.168.12.2 24
int g0/0/1
ip add 192.168.23.2 24
q
```

```
R3:
undo ter mo
sys
sysname R3
user-interface console 0
idle-timeout 0 0
int loop 0
ip add 3.3.3.3 24
int g0/0/1
ip add 192.168.23.3 24
q
```

```
R4:
undo ter mo
```

```
sys
sysname R4
user-interface console 0
idle-timeout 0 0
int loop 0
ip add 4.4.4.4 24
int loop 1
ip add 44.44.44.44 24
int g0/0/0
ip add 192.168.14.4 24
q
```

配置 OSPF IS-IS 协议，R3 环回口通过引入的方向，放到 IS-IS 协议中

```
R1:
ospf router-id 1.1.1.1
area 0
net 192.168.14.1 0.0.0.0
q
```

```
isis
network-entity 10.0000.0000.0001.00
is-name R1
is-level level-1
int g0/0/0
isis enable
q
```

```
R2:
isis
network-entity 10.0000.0000.0002.00
is-name R2
int g0/0/0
```

```
isis enable
isis circuit-level level-1
int g0/0/1
isis enable
isis circuit-level level-2
q
```

```
R3:
isis
network-entity 20.0000.0000.0003.00
is-name R3
is-level level-2
int g0/0/1
isis enable
q
```

```
R4:
ospf router-id 4.4.4.4
area 0
net 192.168.14.4 0.0.0.0
net 4.4.4.4 0.0.0.0
net 44.44.44.44 0.0.0.0
q
```

=====

引入外部直连路由

```
R3:
isis
import-route direct
```

<R3>display isis route

ISIS(1) Level-2 Redistribute Table

Type	IPv4 Destination	IntCost	ExtCost	Tag
D	3.3.3.0/24	0	0	

默认，IS-IS 以 Level-2 路由类型来引入外部路由，被引入路由的 IntCost、ExtCost 值均为 0，开销类型为 External

在 R2 上查看 IS-IS 路由表

IPv4 Destination	IntCost	ExtCost	ExitInterface	NextHop
3.3.3.0/24	10	0	GE0/0/1	192.168.23.3
192.168.23.0/24	10	NULL	GE0/0/1	Direct
192.168.12.0/24	10	NULL	GE0/0/0	Direct

在路由表中，显示为 74

3.3.3.0/24	ISIS-L2	15	74	D	192.168.23.3
127.0.0.0/8	Direct	0	0	D	127.0.0.1
127.0.0.1/32	Direct	0	0	D	127.0.0.1
192.168.12.0/24	Direct	0	0	D	192.168.12.2
192.168.12.2/32	Direct	0	0	D	127.0.0.1
192.168.23.0/24	Direct	0	0	D	192.168.23.2
192.168.23.2/32	Direct	0	0	D	127.0.0.1

对于 External 类型的 cost，在 IP 路由表中的 cost 值的计算方法为，64+ IS-IS 路由表中的 IntCost+ IS-IS 路由表中的 Ex

tCost.

External 的 cost 计算方法：

$64 + \text{InCost} + \text{ExtCost} = 64 + 10 + 0 = 74$

R1 可以 ping 通 R3

<R1>ping 3.3.3.3

[R1]ping 3.3.3.3

PING 3.3.3.3: 56 data bytes, press CTRL_C to break

Reply from 3.3.3.3: bytes=56 Sequence=1 ttl=254 time=120 ms

Reply from 3.3.3.3: bytes=56 Sequence=2 ttl=254 time=60 ms

Reply from 3.3.3.3: bytes=56 Sequence=3 ttl=254 time=70 ms

Reply from 3.3.3.3: bytes=56 Sequence=4 ttl=254 time=60 ms

Reply from 3.3.3.3: bytes=56 Sequence=5 ttl=254 time=60 ms

=====

引入外部 OSPF 路由

在 R1 上以 cost 类型为 Internal 的方式将 OSPF 协议引入到 IS-IS 中，引入时 InCost 改为 30，引入时改为 level-1

R1：

isis

import-route ospf 1 level-1 cost-type internal cost 30

R2 上查看 IS-IS 路由表

Type	IPv4 Destination	IntCost	ExtCost Tag
D	192.168.14.0/24	30	NULL
O	44.44.44.44/32	30	NULL
O	4.4.4.4/32	30	NULL

R2 上能看到引入的路由

=====

使用 route-policy 控制路由的引入
只要求 R4 的 4.4.4.4 引入 IS-IS

R1:
acl 2000
rule permit source 4.4.4.0 0.0.0.255
route-policy 10 permit node 1
if-match acl 2000
isis
import-route ospf 1 route-policy 10 level-1 cost-type
internal cost 30

在 R2 上查看一下，只有 4.4.4.4 引入了

ISIS(1) Level-1 Redistribute Table

Type	IPv4 Destination	IntCost	ExtCost Tag
O	4.4.4.4/32	30	NULL

=====

为了让 R4 与 R3 的环回口可以相互 ping 通，在 R1 上通过 OSPF 下放默认路由

```
R1 :  
ospf  
default-route-advertise
```

检测连通性

```
R4 :  
ping -a 4.4.4.4 3.3.3.3
```

```
[R4]ping -a 4.4.4.4 3.3.3.3  
PING 3.3.3.3: 56 data bytes, press CTRL_C to break  
Reply from 3.3.3.3: bytes=56 Sequence=1 ttl=253 time=90 ms  
Reply from 3.3.3.3: bytes=56 Sequence=2 ttl=253 time=80 ms  
Reply from 3.3.3.3: bytes=56 Sequence=3 ttl=253 time=80 ms  
Reply from 3.3.3.3: bytes=56 Sequence=4 ttl=253 time=60 ms  
Reply from 3.3.3.3: bytes=56 Sequence=5 ttl=253 time=90 ms
```

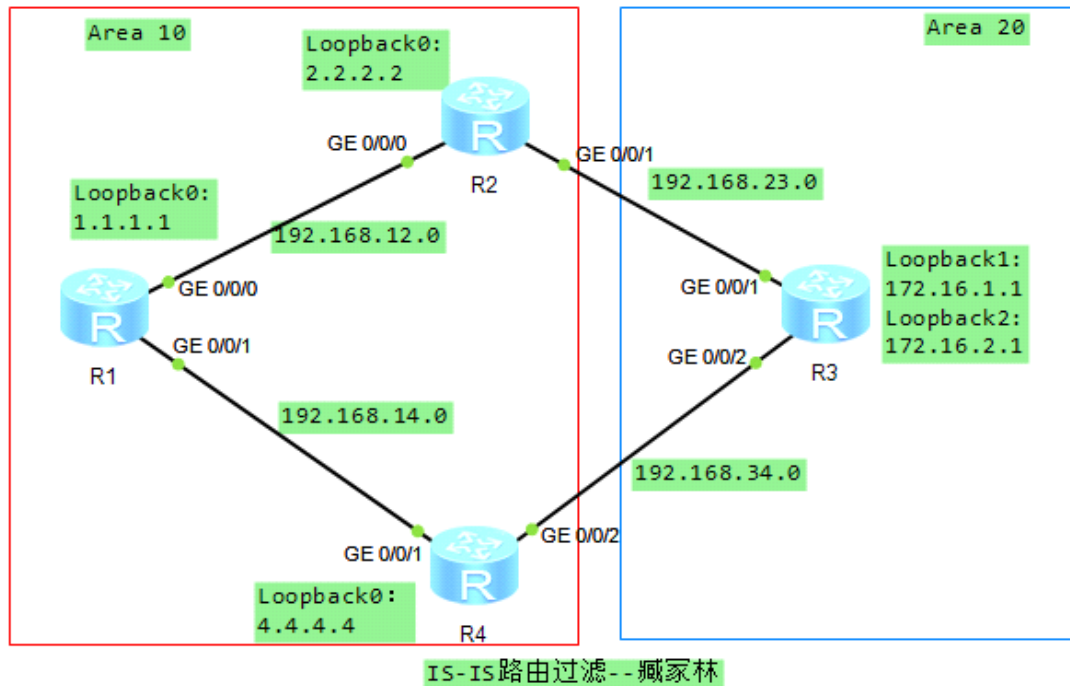
=====

IS-IS 实验 9：IS-IS 路由过滤

在 IS-IS 网络中，有时需要使用 filter-policy 工具对 IS-IS 路由进行过滤。这里所说的过滤，是指路由器在将自己 IS-IS 路由表中的某些 IS-IS 路由纳入进自己的 IP 路由表的过程，一些满足了过滤条件的 IS-IS 路由将被限制纳入 IP 路由表中。

需要注意的是，filter-policy 进行过滤的并非是生成那些 IS-IS 路由的 LSP，所以 filter-policy 进行路由过滤之后，路由器中

的 IS-IS 链路状态数据库和 IS-IS 路由表都不会受到任何影响。



所有路由器均为 level-2 路由器

基本配置

R1:

```
undo ter mo
```

```
sys
```

```
sysname R1
```

```
user-interface console 0
```

```
idle-timeout 0 0
```

```
int loop 0
```

```
ip add 1.1.1.1 24
```

```
int g0/0/0
```

```
ip add 192.168.12.1 24
```

```
int g0/0/1
```

```
ip add 192.168.14.1 24
q
```

```
R2:
undo ter mo
sys
sysname R2
user-interface console 0
idle-timeout 0 0
int loop 0
ip add 2.2.2.2 24
int g0/0/0
ip add 192.168.12.2 24
int g0/0/1
ip add 192.168.23.2 24
q
```

```
R3:
undo ter mo
sys
sysname R3
user-interface console 0
idle-timeout 0 0
int loop 1
ip add 172.16.1.1 24
int loop 2
ip add 172.16.2.1 24
int g0/0/1
ip add 192.168.23.3 24
int g0/0/2
ip add 192.168.34.3 24
q
```

```
R4:
```

```
undo ter mo
sys
sysname R4
user-interface console 0
idle-timeout 0 0
int loop 0
ip add 4.4.4.4 24
int g0/0/1
ip add 192.168.14.4 24
int g0/0/2
ip add 192.168.34.4 24
q
```

配置 IS-IS 协议，所有路由器均为 Level-2 路由器

```
R1:
isis
network-entity 10.0000.0000.0001.00
is-name R1
is-level level-2
int loo 0
isis enable
int g0/0/0
isis enable
int g0/0/1
isis enable
q
```

```
R2:
isis
network-entity 10.0000.0000.0002.00
is-name R2
is-level level-2
int loo 0
isis enable
```

```
int g0/0/0
isis enable
int g0/0/1
isis enable
q
```

```
R3:
isis
network-entity 20.0000.0000.0003.00
is-name R3
is-level level-2
int loo 1
isis enable
int loo 2
isis enable
int g0/0/1
isis enable
int g0/0/2
isis enable
q
```

```
R4:
isis
network-entity 10.0000.0000.0004.00
is-name R4
is-level level-2
int loo 0
isis enable
int g0/0/1
isis enable
int g0/0/2
isis enable
q
```

配置完成后，查看邻居的建立
路由表学习也正常

=====

使用 filter-policy 实现路由过滤

R1 只能经 R2 访问 R3 的 loopback 1

R1 只能经 R4 访问 R3 的 loopback 2

在 R2 上过滤掉 R3 的 loopback 2

R2:

acl 2000

rule deny source 172.16.2.0 0.0.0.255

rule permit source any

isis

filter-policy 2000 import

<R2>display ip routing-table

1.1.1.0/24	ISIS-L2	15	10	D	192.168.12.1
2.2.2.0/24	Direct	0	0	D	2.2.2.2
2.2.2.2/32	Direct	0	0	D	127.0.0.1
4.4.4.0/24	ISIS-L2	15	20	D	192.168.12.1
	ISIS-L2	15	20	D	192.168.23.3
127.0.0.0/8	Direct	0	0	D	127.0.0.1
127.0.0.1/32	Direct	0	0	D	127.0.0.1
172.16.1.0/24	ISIS-L2	15	10	D	192.168.23.3
192.168.12.0/24	Direct	0	0	D	192.168.12.2
192.168.12.2/32	Direct	0	0	D	127.0.0.1

172.16.2.0 已经没有了，只有 1.0

但在 IS-IS 路由表中，还是有的

display isis route

IPv4 Destination	IntCost	ExtCost	ExitInterface

4.4.4.0/24	20	NULL	GE0/0/0 GE0/0/1
192.168.34.0/24	20	NULL	GE0/0/1
172.16.2.0/24	10	NULL	GE0/0/1
192.168.14.0/24	20	NULL	GE0/0/0
172.16.1.0/24	10	NULL	GE0/0/1
192.168.23.0/24	10	NULL	GE0/0/1
2.2.2.0/24	0	NULL	Loop0

过滤掉的路由条目不能进入 IP 路由表，但并不会对生成这些路由的 LSP 进行过滤。

在 R4 上过滤掉 R3 的 loopback 1

R4:

```
acl 2000
```

```
rule deny source 172.16.1.0 0.0.0.255
```

```
rule permit source any
```

```
isis
```

```
filter-policy 2000 import
```

查看一下，R4 的路由表中，已经没有 172.16.1.0 的路由

=====

配合使用 filter-policy 和 route-policy

在 R1 上查看路由表，还是有两个下一跳的，在 R2 R4 上的过滤，对 R1 不起作用，这两条路由的 LSP 还继续泛洪。

在 R1 上进行路由过滤时，不能只考虑路由的前缀，同时还要考虑到路由的下一跳。

R1 :

定义 ACL 2012 匹配路由下一跳 192.168.12.2

定义 ACL 2014 匹配路由下一跳 192.168.14.4

定义 ACL 2001 匹配路由前缀 172.16.1.0/24

定义 ACL 2002 匹配路由前缀 172.16.2.0/24

R1:

acl 2001

rule permit source 172.16.1.0 0.0.0.255

acl 2002

rule permit source 172.16.2.0 0.0.0.255

acl 2012

rule permit source 192.168.12.2 0

acl 2014

rule permit source 192.168.14.4 0

route-policy 10 deny node 1

if-match ip next-hop acl 2012

if-match acl 2002

route-policy 10 deny node 2

if-match ip next-hop acl 2014

if-match acl 2001

route-policy 10 permit node 3

isis

filter-policy route-policy 10 import

=====

也可以 permit 放行

R1 :

```
route-policy 10 permit node 1
if-match ip next-hop acl 2012
if-match acl 2001
```

```
route-policy 10 permit node 2
if-match ip next-hop acl 2014
if-match acl 2002
```

配置完成后，查看 R1 的路由表

1.1.1.0/24	Direct	0	0	D	1.1.1.1
1.1.1.1/32	Direct	0	0	D	127.0.0.1
2.2.2.0/24	ISIS-L2	15	10	D	192.168.12.2
4.4.4.0/24	ISIS-L2	15	10	D	192.168.14.4
127.0.0.0/8	Direct	0	0	D	127.0.0.1
127.0.0.1/32	Direct	0	0	D	127.0.0.1
172.16.1.0/24	ISIS-L2	15	20	D	192.168.12.2
172.16.2.0/24	ISIS-L2	15	20	D	192.168.14.4
192.168.12.0/24	Direct	0	0	D	192.168.12.1
192.168.12.1/32	Direct	0	0	D	127.0.0.1
192.168.14.0/24	Direct	0	0	D	192.168.14.1
192.168.14.1/32	Direct	0	0	D	127.0.0.1

现在完成要求。

= = = = =

IS-IS 实验 10 : IS-IS 路由渗透

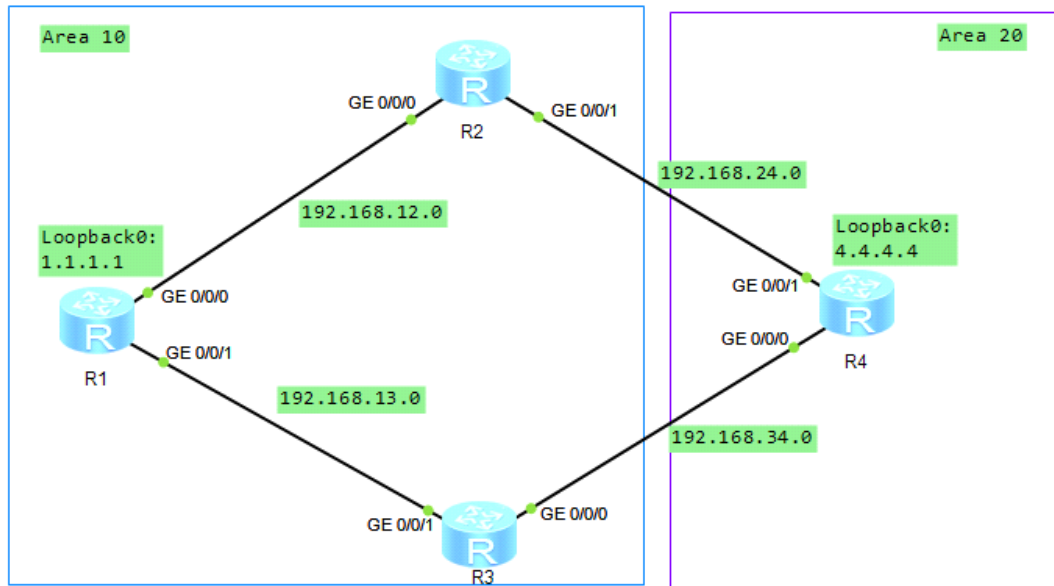
在 IS-IS 网络中，所有的 level-2 和 level-1-2 路由器构成了一个连续的骨干区域。

level-1 区域必须且只能与骨干区域相连，不同 level-1 区域之间不能直接相连。level-1 区域内的路由信息会通过 level-1-2 路由器通报给 level-2 区域，即 level-1-2 路由器会将学习到的 level-1 路由信息封装进 level-2 LSP，并将些 level-2 LSP 传递给其他 level-2 和 level-1-2 路由器。因此，level-1-2 和 level-2 路由器是知道整个 IS-IS 路由域的路由信息的。

另一方面，为了减小路由表的规模，在缺省情况下，level-1-2 和 level-2 路由器并不会将自己知道的路由域中其他 level-1 区域以及骨干区域的路由信息通报给 level-1 区域。这样一来，level-1 路由器只能通过缺省路由来访问来区域以外的任何目的地。

通常情况下，level-1 路由器只通过缺省路由来访问本区域以外的目的地，但是如果需求特殊，则这种方式或许不能被接受。例如，如果要求一个 level-1 路由器必须经由最优路径（也就是总的开销值最小）访问其他某个区域的目的地时，使用缺省路由就无法满足需求。在这种情况下，level-1 路由器需要知道并使用其他区域中的目的地明细路由，而不是盲目地使用缺省路由。

IS-IS 路由渗透指的是 Level 1-2 和 Level-2 路由将自己知道的其他 Level-1 区域以及 Level-2 区域的路由信息通报给指定的 Level-1 区域的过程。



IS-IS路由渗透--臧家林

基本配置

R1:

```
undo ter mo
sys
sysname R1
user-interface console 0
idle-timeout 0 0
int loop 0
ip add 1.1.1.1 24
int g0/0/0
ip add 192.168.12.1 24
int g0/0/1
ip add 192.168.13.1 24
q
```

R2:

```
undo ter mo
sys
sysname R2
user-interface console 0
idle-timeout 0 0
```

```
int g0/0/0
ip add 192.168.12.2 24
int g0/0/1
ip add 192.168.24.2 24
q
```

```
R3:
undo ter mo
sys
sysname R3
user-interface console 0
idle-timeout 0 0
int g0/0/1
ip add 192.168.13.3 24
int g0/0/0
ip add 192.168.34.3 24
q
```

```
R4:
undo ter mo
sys
sysname R4
user-interface console 0
idle-timeout 0 0
int loop 0
ip add 4.4.4.4 24
int g0/0/1
ip add 192.168.24.4 24
int g0/0/0
ip add 192.168.34.4 24
q
```

配置 IS-IS 协议

R1:

```
isis
network-entity 10.0000.0000.0001.00
is-name R1
is-level level-1
int loo 0
isis enable
int g0/0/0
isis enable
int g0/0/1
isis enable
q
```

```
R2:
isis
network-entity 10.0000.0000.0002.00
is-name R2
int g0/0/0
isis circuit-level level-1
isis enable
int g0/0/1
isis enable
isis circuit-level level-2
q
```

```
R3:
isis
network-entity 10.0000.0000.0003.00
is-name R3
int g0/0/0
isis enable
isis circuit-level level-2
int g0/0/1
isis enable
isis circuit-level level-1
```

q

R4:

isis

network-entity 20.0000.0000.0004.00

is-name R4

is-level level-2

int loo 0

isis enable

int g0/0/0

isis enable

int g0/0/1

isis enable

q

配置完成，查看邻居关系的建立

=====

配置 IS-IS 路由渗透

修改 R1 的 g0/0/0 接口 cost 值为 10，g0/0/1 接口 cost 值为 20

R2 的 g0/0/1 接口 cost 值为 30，R3 的 g0/0/0 接口 cost 值为 10

R1:

int g0/0/0

isis cost 10

int g0/0/1

isis cost 20

R2:

```
int g0/0/1
isis cost 30
```

```
R3:
int g0/0/0
isis cost 10
```

配置完成后，R1 验证 1.1.1.1 到 4.4.4.4 所经路径

```
<R1>tracert -a 1.1.1.1 4.4.4.4
```

```
[R1]tracert -a 1.1.1.1 4.4.4.4
```

```
tracert to 4.4.4.4(4.4.4.4), max hops:
o break
```

```
1 192.168.12.2 50 ms 30 ms 50 ms
```

```
2 192.168.24.4 80 ms 60 ms 60 ms
```

```
[R1]
```

可以看到，报文经过的中转路由器是 R2，修改 cost 之后，Level-1 路由器 R1 选择了离它最近（cost 值最小）的 Level-1-2 路由器 R2 作为它自动生成的缺省路由的下一跳，从 1.1.1.1 到 4.4.4.4 报文就是经这条缺省路由发送出去的。

但分析表明，R1 到 R2 到 R4 的总的开销值为 40，而从 R1 到 R3 到 R4 的总的开销值为 30，所以现在的 R1 去 R4 选择的是一条次优路径。

解决方法，在 Level-1-2 路由器 R2 和 R3 上配置路由渗透

```
R2:
```

```
isis
import-route isis level-2 into level-1
```

```
R3:
isis
import-route isis level-2 into level-1
```

配置好这后，R1 上有 4.4.4.4 的明细路由

ation/Mask	Proto	Pre	Cost	Flags	NextHop
0.0.0.0/0	ISIS-L1	15	10	D	192.168.12.2
1.1.1.0/24	Direct	0	0	D	1.1.1.1
1.1.1.1/32	Direct	0	0	D	127.0.0.1
4.4.4.0/24	ISIS-L1	15	40	D	192.168.13.3

现在是从 R3 经过的

```
[R1]tracert -a 1.1.1.1 4.4.4.4
```

```
tracert to 4.4.4.4(4.4.4.4), max hops
o break
```

```
1 192.168.13.3 60 ms 50 ms 40 ms
```

```
2 192.168.34.4 70 ms 40 ms 50 ms
```

```
[R1]
```