

目 录

1 IP 路由基础	1-1
1.1 IP 路由简介	1-1
1.1.1 路由表	1-1
1.1.2 路由分类	1-1
1.1.3 路由协议分类	1-1
1.1.4 路由优先级	1-2
1.1.5 负载分担	1-3
1.1.6 路由备份	1-3
1.1.7 路由迭代	1-3
1.1.8 路由共享	1-3
1.1.9 路由扩展	1-3
1.2 配置路由和标签在 RIB 中的最大存活时间	1-4
1.3 配置路由在 FIB 中的最大存活时间	1-4
1.4 配置 RIB 向 FIB 下发路由时会携带属性消息	1-5
1.5 配置路由的 NSR 功能	1-5
1.6 配置路由由不同协议间快速重路由功能	1-6
1.7 配置直连路由与 Track 项联动	1-7
1.8 配置路由按照路由策略进行迭代下一跳查找	1-9
1.9 配置下一跳循环迭代抑制功能	1-10
1.10 配置设备支持的最大激活路由前缀数	1-11
1.11 开启 MTP 功能	1-12
1.12 路由表显示和维护	1-12

1 IP 路由基础

本手册仅介绍单播路由协议，组播路由协议请参见“IP 组播配置指导”。

1.1 IP路由简介

在网络中路由器根据所收到的报文的地址选择一条合适的路径，并将报文转发到下一个路由器。路径中最后一个路由器负责将报文转发给目的主机。路由就是报文在转发过程中的路径信息，用来指导报文转发。

1.1.1 路由表

RIB（Routing Information Base，路由信息库），是一个集中管理路由信息的数据库，包含路由表信息以及路由周边信息（路由迭代信息、路由共享信息以及路由扩展信息）等。

路由器通过对路由表进行优选，把优选路由下发到 FIB（Forwarding Information Base，转发信息库）表中，通过 FIB 表指导报文转发。FIB 表中每条转发项都指明了要到达某子网或某主机的报文应通过路由器的哪个物理接口发送，就可以到达该路径的下一个路由器，或者不需再经过别的路由器便可传送到直接相连的网络中的目的主机。FIB 表的具体内容，请参见“三层技术-IP 业务配置指导”中的“IP 转发基础”。

1.1.2 路由分类

表1-1 路由分类

分类标准	具体分类
根据来源不同	<ul style="list-style-type: none">直连路由：链路层协议发现的路由，也称为接口路由静态路由：网络管理员手工配置的路由。静态路由配置方便，对系统要求低，适用于拓扑结构简单并且稳定的小型网络。其缺点是每当网络拓扑结构发生变化，都需要手工重新配置，不能自动适应动态路由：路由协议发现的路由
根据路由目的地的不同	<ul style="list-style-type: none">网段路由：目的地为网段，子网掩码长度小于 32 位主机路由：目的地为主机，子网掩码长度为 32 位
根据目的地与该路由器是否直接相连	<ul style="list-style-type: none">直接路由：目的地所在网络与路由器直接相连间接路由：目的地所在网络与路由器非直接相连

1.1.3 路由协议分类

路由协议有自己的路由算法，能够自动适应网络拓扑的变化，适用于具有一定规模的网络拓扑。其缺点是配置比较复杂，对系统的要求高于静态路由，并占用一定的网络资源。

对路由协议的分类可采用以下不同标准。

表1-2 路由协议分类

分类标准	具体分类
根据作用范围	<ul style="list-style-type: none"> IGP (Interior Gateway Protocol, 内部网关协议): 在一个自治系统内部运行, 常见的 IGP 协议包括 RIP、OSPF 和 IS-IS EGP (Exterior Gateway Protocol, 外部网关协议): 运行于不同自治系统之间, BGP 是目前最常用的 EGP
根据使用算法	<ul style="list-style-type: none"> 距离矢量 (Distance-Vector) 协议: 包括 RIP 和 BGP。其中, BGP 也被称为路径矢量协议 (Path-Vector) 链路状态 (Link-State) 协议: 包括 OSPF 和 IS-IS
根据目的地址类型	<ul style="list-style-type: none"> 单播路由协议: 包括 RIP、OSPF、BGP 和 IS-IS 等 组播路由协议: 包括 PIM-SM、PIM-DM 等
根据IP协议版本	<ul style="list-style-type: none"> IPv4 路由协议: 包括 RIP、OSPF、BGP 和 IS-IS 等 IPv6 路由协议: 包括 RIPng、OSPFv3、IPv6 BGP 和 IPv6 IS-IS 等

AS (Autonomous System, 自治系统) 是拥有同一选路策略, 并在同一技术管理部门下运行的一组路由器。

1.1.4 路由优先级

对于相同的目的地, 不同的路由协议、直连路由和静态路由可能会发现不同的路由, 但这些路由并不都是最优的。为了判断最优路由, 各路由协议、直连路由和静态路由都被赋予了一个优先级, 具有较高优先级的路由协议发现的路由将成为最优路由。

除直连路由外, 各路由协议的优先级都可由用户手工进行配置。另外, 每条静态路由的优先级都可以不相同。缺省的路由优先级如表 1-3 所示, 数值越小表明优先级越高。

表1-3 缺省的路由优先级

路由协议或路由种类	缺省的路由优先级
DIRECT (直连路由)	0
组播静态路由	1
OSPF	10
IS-IS	15
单播静态路由	60
RIP	100
OSPF ASE	150
OSPF NSSA	150
IBGP	255
EBGP	255
UNKNOWN (来自不可信源端的路由)	256

1.1.5 负载分担

对同一路由协议来说，允许配置多条目的地相同且开销也相同的路由。当到同一目的地的路由中，没有更高优先级的路由时，这几条路由都被采纳，在转发去往该目的地的报文时，依次通过各条路径发送，从而实现网络的负载分担。

目前支持负载分担有静态路由/IPv6 静态路由、RIP/RIPng、OSPF/OSPFv3、BGP/IPv6 BGP 和 IS-IS/IPv6 IS-IS。

1.1.6 路由备份

使用路由备份可以提高网络的可靠性。用户可根据实际情况，配置到同一目的地的多条路由，其中优先级最高的一条路由作为主路由，其余优先级较低的路由作为备份路由。

正常情况下，路由器采用主路由转发数据。当链路出现故障时，主路由变为非激活状态，路由器选择备份路由中优先级最高的转发数据，实现从主路由到备份路由的切换；当链路恢复正常时，路由器重新选择路由，由于主路由的优先级最高，路由器选择主路由来发送数据，实现从备份路由到主路由的切换。

1.1.7 路由迭代

对于 BGP 路由（直连 EBGP 路由除外）和静态路由（配置了下一跳）以及多跳 RIP 路由而言，其所携带的下一跳信息可能并不是直接可达，需要找到到达下一跳的直连出接口。路由迭代的过程就是通过路由的下一跳信息来找到直连出接口的过程。

而对于 OSPF 和 IS-IS 等链路状态路由协议而言，其下一跳是直接在路由计算时得到的，不需要进行路由迭代。

路由迭代信息记录并保存路由迭代的结果，包括依赖路由的概要信息、迭代路径、迭代深度等。

1.1.8 路由共享

由于各路由协议采用的路由算法不同，不同的路由协议可能会发现不同的路由。如果网络规模较大，当使用多种路由协议时，往往需要在不同的路由协议间能够共享各自发现的路由。

各路由协议都可以引入其它路由协议的路由、直连路由和静态路由，具体内容请参见本手册中各路由协议模块有关引入外部路由的描述。

路由共享信息记录了路由协议之间的引入关系。

1.1.9 路由扩展

路由扩展属性主要是指 BGP 路由的扩展团体属性以及 OSPF 路由的区域 ID、路由类型和 Router ID 等。同路由共享一样，路由协议可以引入其它路由协议的路由扩展属性。

路由扩展信息记录了各路由协议的路由扩展属性以及路由协议扩展属性之间的引入关系。

1.2 配置路由和标签在RIB中的最大存活时间

1. 功能简介

当协议路由表项较多或协议 GR 时间较长时，由于协议收敛速度较慢，可能会出现协议路由表项提前老化的问题。通过调节路由和标签在 RIB 中的最大存活时间，可以解决上面的问题。

2. 配置限制和指导

该配置在下一协议进程倒换或者 RIB 进程倒换时才生效。

3. 配置步骤（IPv4）

- (1) 进入系统视图。

```
system-view
```

- (2) 进入 RIB 视图。

```
rib
```

- (3) 创建 RIB IPv4 地址族，并进入 RIB IPv4 地址族视图。

```
address-family ipv4
```

- (4) 配置 IPv4 路由和标签在 RIB 中的最大存活时间。

```
protocol protocol [instance instance-name] lifetime seconds
```

缺省情况下，IPv4 路由和标签在 RIB 中的最大存活时间为 480 秒。

4. 配置步骤（IPv6）

- (1) 进入系统视图。

```
system-view
```

- (2) 进入 RIB 视图。

```
rib
```

- (3) 创建 RIB IPv6 地址族，并进入 RIB IPv6 地址族视图。

```
address-family ipv6
```

- (4) 配置 IPv6 路由和标签在 RIB 中的最大存活时间。

```
protocol protocol [instance instance-name] lifetime seconds
```

缺省情况下，IPv6 路由和标签在 RIB 中的最大存活时间为 480 秒。

1.3 配置路由在FIB中的最大存活时间

1. 功能简介

当协议进程倒换或 RIB 进程倒换后，如果协议进程没有配置 GR 或 NSR，需要多保留一段时间 FIB 表项；如果协议进程配置了 GR 或 NSR，需要立刻删除 FIB 表项，避免 FIB 表项长时间存在导致问题。通过调节路由在 FIB 中的最大存活时间，可以解决上面的问题。

2. 配置步骤（IPv4）

- (1) 进入系统视图。

```
system-view
```

- (2) 进入 RIB 视图。

rib

- (3) 创建 RIB IPv4 地址族，并进入 RIB IPv4 地址族视图。

address-family ipv4

- (4) 配置 IPv4 路由在 FIB 中的最大存活时间。

fib lifetime seconds

缺省情况下，IPv4 路由在 FIB 中的最大存活时间为 600 秒。

3. 配置步骤（IPv6）

- (1) 进入系统视图。

system-view

- (2) 进入 RIB 视图。

rib

- (3) 创建 RIB IPv6 地址族，并进入 RIB IPv6 地址族视图。

address-family ipv6

- (4) 配置 IPv6 路由在 FIB 中的最大存活时间。

fib lifetime seconds

缺省情况下，IPv6 路由在 FIB 中的最大存活时间为 600 秒。

1.4 配置RIB向FIB下发路由时会携带属性消息

1. 功能简介

配置本功能后，RIB 向 FIB 下刷路由时会携带属性消息。

2. 配置步骤

- (1) 进入系统视图。

system-view

- (2) 进入 RIB 视图。

rib

- (3) 创建 RIB IPv4 地址族，并进入 RIB IPv4 地址族视图。

address-family ipv4

- (4) 配置 RIB 向 FIB 下发路由时携带属性消息。

flush route-attribute protocol

缺省情况下，RIB 向 FIB 下发路由时不携带属性消息。

1.5 配置路由的NSR功能

1. 功能简介

NSR（Nonstop Routing，不间断路由）将路由信息从主进程备份到备进程，在设备发生主备倒换时保证路由信息不丢失，解决了主备倒换期间引发的路由震荡问题，保证转发业务不中断。

路由 NSR 相对于路由协议 NSR 功能，主备倒换时路由收敛速度更快。

2. 配置准备

配置本功能的同时，请配置协议的 GR 或 NSR 功能，否则可能导致路由老化和流量中断。

3. 配置步骤（IPv4）

- (1) 进入系统视图。

```
system-view
```

- (2) 进入 RIB 视图。

```
rib
```

- (3) 创建 RIB IPv4 地址族，并进入 RIB IPv4 地址族视图。

```
address-family ipv4
```

- (4) 配置 IPv4 路由的 NSR 功能。

```
non-stop-routing
```

缺省情况下，IPv4 路由的 NSR 功能处于关闭状态。

4. 配置步骤（IPv6）

- (1) 进入系统视图。

```
system-view
```

- (2) 进入 RIB 视图。

```
rib
```

- (3) 创建 RIB IPv6 地址族，并进入 RIB IPv6 地址族视图。

```
address-family ipv6
```

- (4) 配置 IPv6 路由的 NSR 功能。

```
non-stop-routing
```

缺省情况下，IPv6 路由的 NSR 功能处于关闭状态。

1.6 配置路由不同协议间快速重路由功能

1. 功能简介

当 RIB 表中存在去往同一目的地的多条路由时，路由器会将优先级较高的路由下发到 FIB 表，当该路由的下一跳不可达时，数据流量将会被中断，路由器会重新进行路由优选，优选完毕后，使用新的最优路由来指导报文转发。例如，去往同一个目的地存在一条静态路由和一条 OSPF 路由，缺省情况 OSPF 路由会作为最优路由下发到 FIB 表。当 OSPF 路由的下一跳不可达时，数据流量将会被中断。

通过配置不同协议间快速重路由功能，可以将静态路由的下一跳作为备份下一跳。当路由器检测到网络故障时，将使用备份下一跳替换失效下一跳，通过备份下一跳来指导报文的转发，从而大大缩短了流量中断的时间。

2. 配置限制和指导

使用不同协议间的快速重路由功能生成备份下一跳时可能会造成环路。

3. 配置步骤（IPv4）

- (1) 进入系统视图。

system-view

- (2) 进入 RIB 视图。

rib

- (3) 创建 RIB IPv4 地址族，并进入 RIB IPv4 地址族视图。

address-family ipv4

- (4) 配置 IPv4 路由不同协议间快速重路由功能。

inter-protocol fast-reroute [vpn-instance vpn-instance-name]

缺省情况下，不同协议间快速重路由处于关闭状态。

不指定 VPN 时，开启公网的不同协议间快速重路由功能。

4. 配置步骤（IPv6）

- (1) 进入系统视图。

system-view

- (2) 进入 RIB 视图。

rib

- (3) 创建 RIB IPv6 地址族，并进入 RIB IPv6 地址族视图。

address-family ipv6

- (4) 配置 IPv6 路由不同协议间快速重路由功能。

inter-protocol fast-reroute [vpn-instance vpn-instance-name]

缺省情况下，不同协议间快速重路由处于关闭状态。

不指定 VPN 时，开启公网的不同协议间快速重路由功能。

1.7 配置直连路由与Track项联动

1. 功能简介

在采用 VRRP 备份组作为用户网关的场景中，由于上行流量的路径取决于 VRRP 状态，而下行流量的路径一般由动态路由协议通过选路决定，因此可能出现上下行流量路径不一致的情况。当需要在用户主机上设置防火墙以提高安全性时，上下行路径不一致会导致流量被防火墙阻止；另外，在进行流量监控、流量统计等管理工作时，上下行流量路径不一致会增加不必要的难度和成本。为了解决该问题，可以通过 Track 项在 VRRP 备份组和直连路由之间建立关联，使用户网关设备根据 VRRP 状态来调整直连路由的开销值，然后配置动态路由协议引入直连路由，从而使 VRRP 状态影响动态路由协议的选路结果。具体的工作机制如下：

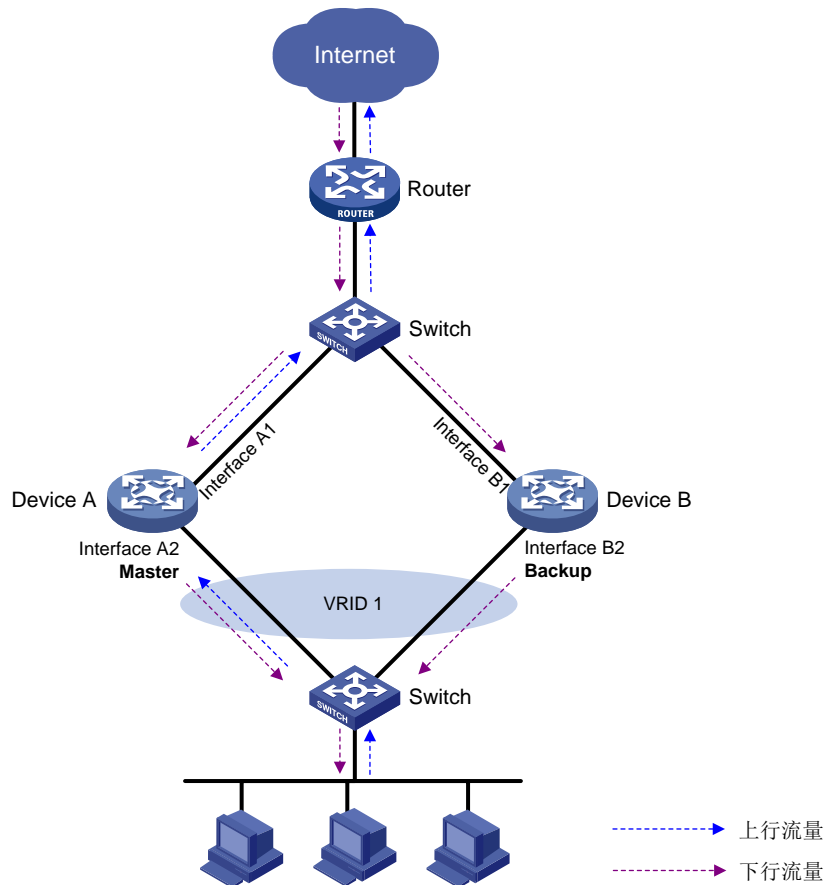
- 如果直连路由关联的 Track 项不存在，或者关联的 Track 项的状态为 NotReady 或 Positive 时，接口的直连路由的开销值为 0，该路由被优选。
- 如果直连路由关联的 Track 项的状态为 Negative 时，该接口的直连路由的开销值为用户通过本功能设置的值，该路由不会被优选。

Track 项的状态与 VRRP 备份组中设备状态的对应关系如下：

- 当设备为 Master 状态时，关联的 Track 项为 Positive 状态。
- 当设备为 Backup 或 Initialize 状态时，关联的 Track 项为 Negative 状态；当设备为 Inactive 状态或 VRRP 备份组不存在时，对应的 Track 项为 NotReady 状态。

如图 1-1 所示,各设备之间通过 OSPF 进行互通。Device A 和 Device B 加入 VRRP 备份组 1, Device A 在 VRRP 备份组 1 中的优先级最高, 正常情况下, 由 Device A 为主机提供网关服务。Router 上分别通过 Device A 和 Device B 去往目的主机的两条路由形成等价路由。上行流量通过 Device A 转发, 下行流量通过等价路由进行负载分担, 导致上下行流量路径不一致。

图1-1 VRRP 备份组作为用户网关的场景



为了避免上述问题的产生, 需要执行如下配置:

- 在 Device A 的 Interface A2 接口下将直连路由与 Track 项关联, 并配置 OSPF 引入直连路由, 且使用该直连路由的原有开销值。
- 在 Device B 的 Interface B2 接口下将直连路由与 Track 项关联, 并配置 OSPF 引入直连路由, 且使用该直连路由的原有开销值。

完成上述配置后, 由于 Device B 的状态为 Backup, 那么 VRRP 1 关联的 Track 项的状态为 Negative, Device B 生成的接口 Interface B2 路由的开销值由 0 变为用户设置的较大值, Router 经过路由计算后选择通过 Device A 去往目的主机, 从而保证上下行流量路径一致。

关于 VRRP 备份组关联 Track 项的详细介绍, 请参见“可靠性配置指导”中的“Track”。

2. 配置限制和指导

不能通过重复执行 `route-direct track` 命令修改直连路由关联的 Track 项。如需修改直连路由关联的 Track 项, 请先通过 `undo route-direct track` 命令取消直连路由关联的 Track 项, 再执行本命令。

3. 配置步骤

- (1) 进入系统视图。

```
system-view
```

- (2) 创建与备份组关联的 Track 项，并进入 Track 视图。

```
track track-entry-number vrrp interface interface-type  
interface-number vrid virtual-router-id
```

关于 Track 与 VRRP 联动的详细介绍，请参见“可靠性配置指导”中的“Track”。

- (3) 退回系统视图。

```
quit
```

- (4) 进入接口视图。

```
interface interface-type interface-number
```

- (5) 配置直连路由与 Track 项关联。

```
route-direct track track-entry-number degrade-cost cost-value
```

缺省情况下，未配置直连路由与 Track 项关联。

1.8 配置路由按照路由策略进行迭代下一跳查找

1. 功能简介

通过配置按路由策略迭代下一跳，可以对路由迭代的结果进行控制。例如：当路由发生变化时，路由管理需要对非直连的下一跳重新进行迭代。如果不对迭代的结果路由进行任何限制，则路由管理可能会将下一跳迭代到一个错误的转发路径上，从而造成流量丢失。此时，可以通过配置本功能，将错误的依赖路由过滤掉，使路由迭代到通过路由策略过滤的指定依赖路由上。

2. 配置限制和指导

配置路由策略时，如果配置了 apply 子句，apply 子句不会生效。

配置路由策略时，请确保至少有一个正确的依赖路由能够通过该策略的过滤，否则可能导致相关路由不可达，无法正确指导转发。

3. 配置步骤（IPv4 单播）

- (1) 进入系统视图。

```
system-view
```

- (2) 进入 RIB 视图。

```
rib
```

- (3) 创建 RIB IPv4 地址族，并进入 RIB IPv4 地址族视图。

```
address-family ipv4
```

- (4) 配置路由按照路由策略进行迭代下一跳查找。

```
protocol protocol nexthop recursive-lookup route-policy  
route-policy-name
```

缺省情况下，未配置路由按路由策略进行下一跳迭代查找。

4. 配置步骤（IPv6 单播）

- (1) 进入系统视图。

```
system-view
```

- (2) 进入 RIB 视图。

```
rib
```

- (3) 创建 RIB IPv6 地址族，并进入 RIB IPv6 地址族视图。

```
address-family ipv6
```

- (4) 配置路由按照路由策略进行迭代下一跳查找。

```
protocol protocol nexthop recursive-lookup route-policy  
route-policy-name
```

缺省情况下，未配置路由按路由策略进行下一跳迭代查找。

1.9 配置下一跳循环迭代抑制功能

1. 功能简介

在路由迭代过程中，若迭代路径包含该路由本身，则认为发生循环迭代。循环迭代将导致本次迭代失败并触发继续查找其他依赖路由。当大量路由的下一跳相同，且该下一跳因不断进行循环迭代导致路由频繁震荡时，系统会频繁处理大量的路由变化，这样会占用大量系统资源，导致 CPU 占用率升高，影响设备性能。

不断进行循环迭代时，迭代失败次数（下一跳循环迭代惩罚计数）会累计一次，当惩罚计数达到 20 次时，将不再进行路由迭代，从而解决上述问题。

2. 配置限制和指导

关闭循环迭代抑制的配置会立即生效。

清除惩罚计数时间间隔的配置在下次进入迭代抑制状态时生效。

3. 关闭路由下一跳循环迭代抑制功能（IPv4）

- (1) 进入系统视图。

```
system-view
```

- (2) 进入 RIB 视图。

```
rib
```

- (3) 创建 RIB IPv4 地址族，并进入 RIB IPv4 地址族视图。

```
address-family ipv4
```

- (4) 关闭 IPv4 路由下一跳循环迭代抑制功能。

```
nexthop recursive-lookup restrain disable
```

缺省情况下，下一跳 IPv4 路由循环迭代抑制功能处于开启状态。

用户不关心循环迭代造成的 CPU 占用率升高时，可以关闭下一跳循环迭代抑制功能。

4. 配置清除下一跳循环迭代惩罚计数的时间间隔（IPv4）

- (1) 进入系统视图。

```
system-view
```

- (2) 进入 RIB 视图。

```
rib
```

- (3) 创建 RIB IPv4 地址族，并进入 RIB IPv4 地址族视图。

```
address-family ipv4
```

- (4) 配置 IPv4 路由清除下一跳循环迭代惩罚计数的时间间隔。

```
nexthop recursive-lookup restrain clear-interval interval
```

缺省情况下，IPv4 路由清除下一跳循环迭代惩罚计数的时间间隔为 600 秒。

5. 关闭路由下一跳循环迭代抑制功能（IPv6）

- (1) 进入系统视图。

```
system-view
```

- (2) 进入 RIB 视图。

```
rib
```

- (3) 创建 RIB IPv6 地址族，并进入 RIB IPv6 地址族视图。

```
address-family ipv6
```

- (4) 关闭 IPv6 路由下一跳循环迭代抑制功能。

```
nexthop recursive-lookup restrain disable
```

缺省情况下，IPv6 路由下一跳循环迭代抑制功能处于开启状态。

用户不关心循环迭代造成的 CPU 占用率升高时，可以关闭下一跳循环迭代抑制功能。

6. 配置清除下一跳循环迭代惩罚计数的时间间隔（IPv6）

- (1) 进入系统视图。

```
system-view
```

- (2) 进入 RIB 视图。

```
rib
```

- (3) 创建 RIB IPv6 地址族，并进入 RIB IPv6 地址族视图。

```
address-family ipv6
```

- (4) 配置 IPv6 路由清除下一跳循环迭代惩罚计数的时间间隔。

```
nexthop recursive-lookup restrain clear-interval interval
```

缺省情况下，IPv6 路由清除下一跳循环迭代惩罚计数的时间间隔为 600 秒。

1.10 配置设备支持的最大激活路由前缀数

1. 功能简介

配置设备支持的最大 IPv4/IPv6 激活路由前缀数后，当设备上的 IPv4/IPv6 激活路由前缀数超过最大支持的激活路由前缀数目时，可以继续激活新的路由前缀，但会产生一条日志信息提示用户，以使用户及时执行必要的操作，以免 IPv4/IPv6 激活路由前缀占用过多的资源。

2. 配置步骤（IPv4）

- (1) 进入系统视图。

```
system-view
```

- (2) 进入 RIB 视图。

```
rib
```

- (3) 创建 RIB IPv4 地址族，并进入 RIB IPv4 地址族视图。

```
address-family ipv4
```

- (4) 配置最大 IPv4 激活路由前缀数。

```
routing-table limit number simply-alert
```

缺省情况下，不限制设备支持的最大 IPv4 激活路由前缀数。

RIB IPv4 地址族视图下的配置用于控制公网和所有 VPN 实例内 IPv4 激活路由的总数。

3. 配置步骤（IPv6）

- (1) 进入系统视图。

```
system-view
```

- (2) 进入 RIB 视图。

```
rib
```

- (3) 创建 RIB IPv6 地址族，并进入 RIB IPv6 地址族视图。

```
address-family ipv6
```

- (4) 配置最大 IPv6 激活路由前缀数。

```
routing-table limit number simply-alert
```

缺省情况下，不限制设备支持的最大 IPv6 激活路由前缀数。

RIB IPv6 地址族视图下的配置用于控制公网和所有 VPN 实例内 IPv6 激活路由的总数。

1.11 开启MTP功能

1. 功能简介

MTP（Maintenance Probe，维护探测器）功能主要用来辅助路由协议实现故障定位，用户可根据网络维护的需要选择是否开启 MTP 功能。开启 MTP 功能后，当与邻居间的路由协议报文超时，设备会向超时的邻居发起 Ping 操作和 Tracert 操作并记录结果，可以通过 **display bgp troubleshooting** 等路由协议模块的显示命令查看 Ping 结果，详细结果信息可通过 **display logbuffer** 命令查看。

2. 配置步骤

- (1) 进入系统视图。

```
system-view
```

- (2) 开启 MTP 功能。

```
maintenance-probe enable
```

缺省情况下，MTP 功能处于关闭状态。

1.12 路由表显示和维护

在任意视图下执行 **display** 命令可以显示路由表信息。在用户视图下执行 **reset** 命令可以清除路由表的统计信息。

表1-4 路由表显示和维护

操作	命令
显示路由表的信息	display ip routing-table [<i>all-vpn-instance</i> <i>vpn-instance vpn-instance-name</i>] [<i>verbose</i>] display ip routing-table [<i>all-routes</i>]
显示通过指定基本访问控制列表过滤的路由信息	display ip routing-table [<i>vpn-instance vpn-instance-name</i>] acl <i>ipv4-acl-number</i> [<i>verbose</i>]
显示指定目的地址的路由	display ip routing-table [<i>vpn-instance vpn-instance-name</i>] <i>ip-address</i> [<i>mask-length</i> <i>mask</i>] [<i>longer-match</i>] [<i>verbose</i>]
显示指定目的地址范围内的路由	display ip routing-table [<i>vpn-instance vpn-instance-name</i>] <i>ip-address1</i> to <i>ip-address2</i> [<i>verbose</i>]
显示通过指定前缀列表过滤的路由信息	display ip routing-table [<i>vpn-instance vpn-instance-name</i>] prefix-list <i>prefix-list-name</i> [<i>verbose</i>]
显示指定协议生成或发现的路由信息	display ip routing-table [<i>vpn-instance vpn-instance-name</i>] protocol <i>protocol</i> [<i>inactive</i> <i>verbose</i>]
显示路由表中路由或前缀的统计信息	display ip routing-table [<i>all-routes</i> <i>all-vpn-instance</i> <i>vpn-instance vpn-instance-name</i>] [<i>prefix</i>] statistics
显示路由表的概要信息	display ip routing-table [<i>vpn-instance vpn-instance-name</i>] summary
显示IPv6 RIB的路由属性信息	display ipv6 rib attribute [<i>attribute-id</i>]
显示IPv6 RIB的GR状态信息	display ipv6 rib graceful-restart
显示IPv6 RIB的下一跳信息	display ipv6 rib nib [<i>self-originated</i>] [<i>nib-id</i>] [<i>verbose</i>] display ipv6 rib nib protocol <i>protocol</i> [<i>verbose</i>]
显示IPv6直连路由下一跳信息	display ipv6 route-direct nib [<i>nib-id</i>] [<i>verbose</i>]
显示IPv6路由表中路由或前缀的统计信息	display ipv6 routing-table [<i>all-routes</i> <i>all-vpn-instance</i> <i>vpn-instance vpn-instance-name</i>] [<i>prefix</i>] statistics
显示IPv6路由表的信息	display ipv6 routing-table [<i>all-vpn-instance</i> <i>vpn-instance vpn-instance-name</i>] [<i>verbose</i>] display ipv6 routing-table [<i>all-routes</i>]
显示通过指定基本IPv6 ACL过滤的IPv6路由信息	display ipv6 routing-table [<i>vpn-instance vpn-instance-name</i>] acl <i>ipv6-acl-number</i> [<i>verbose</i>]
显示指定目的地址的IPv6路由信息	display ipv6 routing-table [<i>vpn-instance vpn-instance-name</i>] <i>ipv6-address</i> [<i>prefix-length</i>] [<i>longer-match</i>] [<i>verbose</i>]
显示指定目的地址范围内的IPv6路由信息	display ipv6 routing-table [<i>vpn-instance vpn-instance-name</i>] <i>ipv6-address1</i> to <i>ipv6-address2</i> [<i>verbose</i>]
显示通过指定前缀列表过滤的IPv6路由信息	display ipv6 routing-table [<i>vpn-instance vpn-instance-name</i>] prefix-list <i>prefix-list-name</i> [<i>verbose</i>]
显示指定协议生成或发现的IPv6路由信息	display ipv6 routing-table [<i>vpn-instance vpn-instance-name</i>] protocol <i>protocol</i> [<i>inactive</i> <i>verbose</i>]
显示IPv6路由表的概要信息	display ipv6 routing-table [<i>vpn-instance vpn-instance-name</i>] summary

操作	命令
显示RIB的路由属性信息	display rib attribute [<i>attribute-id</i>]
显示RIB的GR状态信息	display rib graceful-restart
显示RIB的下一跳信息	display rib nib [<i>self-originated</i>] [<i>nib-id</i>] [<i>verbose</i>] display rib nib protocol <i>protocol</i> [<i>verbose</i>]
显示直连路由下一跳信息	display route-direct nib [<i>nib-id</i>] [<i>verbose</i>]
清除路由表中的综合路由统计信息	reset ip routing-table statistics protocol [<i>vpn-instance</i> <i>vpn-instance-name</i>] { <i>protocol</i> all } reset ip routing-table [all-routes all-vpn-instance] statistics protocol { <i>protocol</i> all }
清除IPv6路由表中的综合路由统计信息	reset ipv6 routing-table statistics protocol [<i>vpn-instance</i> <i>vpn-instance-name</i>] { <i>protocol</i> all } reset ipv6 routing-table [all-routes all-vpn-instance] statistics protocol { <i>protocol</i> all }