Content

Chapter 1 Routing Protocol Overview

To communicate with a remote host over the Internet, a host must choose a proper route via a set of routers or Layer3 switches.

Both routers and layer3 switches calculate the route using CPU, the difference is that layer3 switch adds the calculated route to the switch chip and forward by the chip at wire speed, while the router always store the calculated route in the route table or route buffer, and data forwarding is performed by the CPU. For this reason, although both routers and switches can perform route selection, layer3 switches have great advantage over routers in data forwarding. The following describes basic principle and methods used in layer3 switch route selection.

In route selection, the responsibility of each layer3 switch is to select a proper midway route according to the destination of the packet received; and send the packet to the next layer3 switch until the last layer3 switch in the route send the packet to the destination host. A route is the path selected by each layer3 switch to pass the packet to the next layer3 switch. Route can be grouped into direct route, static route and dynamic route.

Direct route refer to the path directly connects to the layer3 switch, and can be obtained with no calculation.

Static route is the manually specified path to a network or a host; static route cannot be changed freely. The advantage of static route is simple and consistent, and it can limit illegal route modification, and is convenient for load balance and route backup. However, as this is set manually, it is not suitable for mid- or large-scale networks for the route in such conditions are too huge and complex.

Dynamic route is the path to a network or a host calculated by the layer3 switch according to the routing protocols enabled. If the next hop layer3 switch in the path is not reachable, layer3 switch will automatically discard the path to that next hop layer3 switch and choose the path through other layer3 switches.

There are two dynamic routing protocols: Interior Gateway Protocol (IGP) and Exterior Gateway protocol (EGP). IGP is the protocol used to calculate the route to a destination inside an autonomous system. IGP supported by switch include RIP and OSPF, RIP and OSRF can be configured according to the requirement. Switch supports running several IGP dynamic routing protocols at the same time. Or, other dynamic routing protocols and static route can be introduced to a dynamic routing protocol, so that multiple routing protocols can be associated.

EGP is used to exchange routing information among different autonomous systems, such as BGP protocol. EGP supported by switch include BGP-4, BGP-4+.

1.1 Routing Table

As mentioned before, layer3 switch is mainly used to establish the route from the current layer3 switch to a network or a host, and to forward packets according to the route. Each layer3 switch has its own route table containing all routes used by that switch. Each route entry in the route table specifies the physical port should be used for forwarding packet to reach a destination host or the next hop layer3 switch to the host.

The route table mainly consists of the following:

- Destination address: used to identify the destination address or destination network of an IP packet.
- Network mask: used together with destination address to identify the destination host or the network the layer3 switch resides. Network mask consists of several consecutive binary 1's, and usually in the format of dotted decimal (an address consists of 1 to 4 255's.) When "AND" the destination address with network mask, we can get the network address for the destination host or the network the layer3 switch resides. For example, the network address of a host or the segment the layer3 switch resides with a destination address of 200.1.1.1 and mask 255.255.255.0 is 200.1.1.0.
- Output interface: specify the interface of layer3 switch to forward IP packets.
- IP address of the next layer3 switch (next hop): specify the next layer3 switch the IP packet will pass.
- Route entry priority: There may be several different next hop routes leading to the same destination. Those routes may be discovered by different dynamic routing protocols or static routes manually configured. The entry with the highest priority (smallest value) becomes the current best route. The user can configure several routes of different priority to the same destination; layer3 switch will choose one route for IP packet forwarding according to the priority order.

To prevent too large route table, a default route can be set. Once route table look up fails, the default route will be chosen for forwarding packets.

The table below describes the routing protocols supported by switch and the default route look up priority value.

Routing Protocols or route type	Default priority value
Direct route	0
OSPF	110
Static route	1
RIP	120
OSPF ASE	150
IBGP	200

EBGP	20
Unknown route	255

1.2 IP Routing Policy

1.2.1 Introduction to Routing Policy

Some policies have to be applied when the router publishing and receiving routing messages so to filter routing messages, such as only receiving or publishing routing messages meets the specified conditions. A routing protocol maybe need redistribute other routing messages found by other protocols such as OSPF so to increase its own routing knowledge; when the router redistributing routing messages from other routing protocols there may be only part of the qualified routing messages is needed, and some properties may have to be configured to suit this protocol.

To achieve routing policy, first we have to define the characteristics of the routing messages to be applied with routing policies, namely define a group matching rules. We can configure by different properties in the routing messages such as destination address, the router address publishing the routing messages. The matching rules can be previously configured to be applied in the routing publishing, receiving and distributing policies.

Five filters are provided in switch: route-map, acl, as-path, community-list and ip-prefix for use. We will introduce each filter in following sections:

1. route-map

For matching certain properties of the specified routing information and setting some routing propertities when the conditions are fulfilled.

Route-map is for controlling and changing the routing messages while also controlling the redistribution among routes. A route-map consists of a series of match and set commands in which the match command specifies the conditions required matching, and the set command specifies the actions to be taken when matches. The route-map is also for controlling route publishing among different route process. It can also used on policy routing which select different routes for the messages other than the shortest route.

A group matches and set clauses make up a node. A route-map may consist of several nodes each of which is a unit for matching test. We match among nodes with by sequence-number. Match clauses define matching rules. The matching objects are some properties of routing messages. Different match clause in the same node is “and” relation logically, which means the matching test of a node, will not be passed until conditions in its entire match clause are matched. Set clause specifies actions, namely configure some properties of routing messages after the matching test is passed.

Different nodes in a route-map is an “or” relation logically. The system checks each

node of the route-map in turn and once certain node test is passed the route-map test will be passed without taking the next node test.

2. access control list(acl)

ACL (Access Control Lists) is a data packet filter mechanism in the switch. The switch controls the network access and secure the network service by permitting or denying certain data packet transmtting out from or into the network. Users can establish a group of rules by certain messages in the packet, in which each rule to be applied on certain amount of matching messages: permit or deny. The users can apply these rules to the entrance or exit of specified switch, with which data stream in certain direction on certain port would have to follow the specified ACL rules in-and-out the switch. Please refer to chapter “ACL Configuration”.

3. Ip-prefix list

The ip-prefix list acts similarly to acl while more flexible and more understandable. The match object of ip-prefix is the destination address messages field of routing messages when applied in routing messages filtering.

An ip-prefix is identified by prefix list name. Each prefix list may contain multiple items, each of which specifies a matching range of a network prefix type and identifies with a sequence-number which specifies the matching check order of ip-prefix.

In the process of matching, the switch check each items identified by sequence-number in ascending order and the filter will be passed once certain items is matched(without checking rest items)

4. Autonomic system path information access-list as-path

The autonomic system path information access-list as-path is only used in BGP. In the BGP routing messages packet there is an autonomic system path field (in which autonomic system path the routing messages passes through is recorded). As-path is specially for specifying matching conditions for autonomic system path field.

As for relevant as-path configurations, please refer to the ip as-path command in BGP configuration.

5. community-list

Community-list is only for BGP. There is a community property field in the BGP routing messages packet for identifying a community. The community list is for specifying matching conditions for Community-list field.

As for relevant Community-list configuration, please refer to the ip as-path command in BGP configuration

1.2.2 IP Routing Policy Configuration Task List

1． Define route-map

2． Define the match clause in route-map

3. Define the set clause in route-map
4. Define address prefix list

1. Define route-map

Command	Explanation
Global mode	
route-map <*map_name*> {deny | permit} <*sequence_num*> **no route-map <*map_name*> [{deny | permit} <*sequence_num*>]**	Configure route-map; the **no route-map <*map_name*> [{deny | permit} <*sequence_num*>]** command deletes the route-map.

2. Define the match clause in route-map

Command	Explanation
Route-map configuration mode	
match as-path <*list-name*> **no match as-path [<*list-name*>]**	Match the autonomous system as path access-list the BGP route passes through; the **no match as-path [<*list-name*>]** command deletes match condition.
match community <*community-list-name* | *community-list-num* > [exact-match] **no match community [<*community-list-name* | *community-list-num* > [exact-match]]**	Match a community property access-list. The **no match community [<*community-list-name* | *community-list-num* > [exact-match]]** command deletes match condition.
match interface <*interface-name* > **no match interface [<*interface-name* >]**	Match by ports; The **no match interface [<*interface-name* >]** command deletes match condition.

match ip <*address* | *next-hop*> <*ip-acl-name* | *ip-acl-num* | *prefix-list list-name*> **no match ip <*address* | *next-hop*> [<*ip-acl-name* | *ip-acl-num* | *prefix-list [list-name]*>]**	Match the address or next-hop; The **no match ip <address | next-hop> [<*ip-acl-name* | *ip-acl-num* | prefix-list [*list-name*]>]** command deletes match condition.
match metric <*metric-val* > **no match metric [<*metric-val* >]**	Match the routing metric value; The **no match metric [<*metric-val* >]** command deletes match condition.
match origin <*egp* | *igp* | *incomplete* > **no match origin [<*egp* | *igp* | *incomplete* >]**	Match the route origin; The **no match origin [<egp | igp | incomplete >]** command deletes match condition.
match route-type external <*type-1* | *type-2* > **no match route-type external [<*type-1* | *type-2* >]**	Match the route type; The **no match route-type external [<type-1 | type-2 >]** command deletes match condition.
match tag <*tag-val* > **no match tag [<*tag-val* >]**	Match the route tag; The **no match tag [<*tag-val* >]** command deletes match condition.

3. Define the set clause in route-map

Command	Explanation
Route-map configuration mode	
set aggregator as <*as-number*> <*ip_addr*> **no set aggregator as [<*as-number*> <*ip_addr*>]**	Distribute an AS No. for BGP aggregator; The no command deletes the configuration

set as-path prepend <*as-num*> **no set as-path prepend [<*as-num*>]**	Add a specified AS No. before the BGP routing messages as-path series; The no command deletes the configuration
set atomic-aggregate **no set atomic-aggregate**	Configure the BGP atomic aggregate property; The no command deletes the configuration
set comm-list <*community-list-name* | *community-list-num* > delete **no set comm-list <*community-list-name* | *community-list-num* > delete**	Delete BGP community list value; The no command deletes the configuration
set community [AA:NN] [internet] [local-AS] [no-advertise] [no-export] [none] [additive] **no set community [AA:NN] [internet] [local-AS] [no-advertise] [no-export] [none] [additive]**	Configure BGP community list value; The no command deletes the configuration
set extcommunity <*rt* | *soo*> <*AA:NN*> **no set extcommunity <*rt* | *soo*> [<*AA:NN*>]**	Configure BGP extended community list property; The no command deletes the configuration
set ip next-hop <*ip_addr*> **no set ip next-hop [<*ip_addr*>]**	Set next-hop IP address; The no command deletes the configuration
set local-preference <*pre_val*> **no set local-preference [<*pre_val*>]**	Set local preference; The no command deletes the configuration
set metric < +/- *metric_val* | *metric_val*> **no set metric [<+/- *metric_val* | *metric_val*>]**	Set routing metric value; The no command deletes the configuration
set metric-type <*type-1* | *type-2*> **no set metric-type [<*type-1* | *type-2*>]**	Set OSPF metric type; The no command deletes the configuration
set origin <*egp* | *igp* | *incomplete* > **no set origin [<*egp* | *igp* | *incomplete* >]**	Set BGP routing origin; The no command deletes the configuration

set originator-id <*ip_addr*> **no set originator-id [<*ip_addr*>]**	Set routing originator ID; The no command deletes the configuration
set tag <*tag_val*> **no set tag [<*tag_val*>]**	Set OSPF routing tag value; The no command deletes the configuration
set vpnv4 next-hop <*ip_addr*> **no set vpnv4 next-hop [<*ip_addr*>]**	Set BGP VPNv4 next-hop address; the no command deletes the configuration
set weight < *weight_val*> **no set weight [<*weight_val*>]**	Set BGP routing weight; The no command deletes the configuration

4. Define address prefix list

Command	Explanation
Global mode	
ip prefix-list <*list_name*> description <*description*> **no ip prefix-list <*list_name*> description**	Describe the prefix list; The **no ip prefix-list <*list_name*> description** command deletes the configuration.
ip prefix-list <*list_name*> [seq <*sequence_number*>] <*deny* | *permit*> < *any* | *ip_addr/mask_length [ge min_prefix_len] [le max_prefix_len]*> **no ip prefix-list <*list_name*> [seq <*sequence_number*>] [<*deny* | *permit*> < *any* | *ip_addr/mask_length [ge min_prefix_len] [le max_prefix_len]*>]**	Set the prefix list; The **no ip prefix-list <*list_name*> [seq <*sequence_number*>] [<deny | permit> < any | *ip_addr/mask_length* [ge *min_prefix_len*] [le *max_prefix_len*]>]** command deletes the configuration.

1.2.3 Configuration Examples

The figure below shows a network consisting of four Layer 3 switches. This example demonstrates how to set the BGP as-path properties through route-map. BGP protocol is applied among the Layer 3 switches. As for switchC, the network 192.68.11.0/24 can be reached through two paths in which one is AS-PATH 1 by IBGP (going through SwitchD),

the other one is AS-PATH 2 by EBGP (going through SwitchB). BGP selects the shortest path, so AS-PATH 1 is the preferred path. If the path 2 is wished, which is through EBGP path, we can add two extra AS path numbers into the AS-PATH messages from SwitchA to SwitchD so as to change the determination SwitchC take to 192.68.11.0/24.

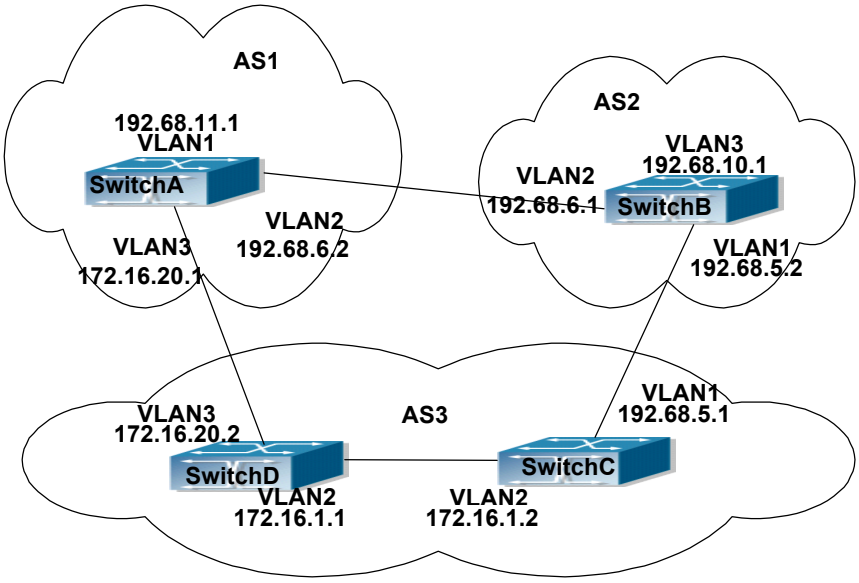


Fig 1-1 Policy routing Configuration

Configuration procedure: (only SwitchA is listed, configurations for other switches are omitted.)

The configuration of Layer 3 switchA:

SwitchA#config

SwitchA(config) #router bgp 1

SwitchA(config-router)#network 192.68.11.0 mask 255.255.255.0

SwitchA(config-router)#neighbor 172.16.20.2 remote-as 3

SwitchA(config-router)#neighbor 172.16.20.2 route-map AddAsNumbers out

SwitchA(config-router)#neighbor 192.68.6.1 remote-as 2

SwitchA(config-router)#exit

SwitchA(config)#route-map AddAsNumbers permit 10

SwitchA(config-route-map)#set as-path prepend 1 1

1.2.4 Troubleshooting

Faq: The routing protocol could not achieve the routing messages study under normal protocol running state

Troubleshooting: check following errors:

- ☞ Each node of route-map should at least has one node is permit match mode. When the route map is used in routing messages filtering, the routing messages will be

considered not pass the routing messages filtering if certain routing messages does not pass the filtering of any nodes. When all nodes are set to deny mode, all routing messages will not pass the filtering in this route-map.

☞ Items in address prefix list should at least have one item set to permit mode. The deny mode items can be defined first to fast remove the unmatched routing messages, however if all the items are set to deny mode, any route will not be able to pass the filtering of this address prefix list. We can define a permit 0.0.0.0/0 le 32 item after several deny mode items are defined so to permit all other routing messages pass through. Only default route will be matched in less-equal 32 is not specified.

Chapter 2 Static Route

2.1 Introduction to Static Route

As mentioned earlier, the static route is the manually specified path to a network or a host. Static route is simple and consistent, and can prevent illegal route modification, and is convenient for load balance and route backup. However, it also has its own defects. Static route, as its name indicates, is static, it won't modify the route automatically on network failure, and manual configuration is required on such occasions, therefore it is not suitable for mid and large-scale networks.

Static route is mainly used in the following two conditions: 1) in stable networks to reduce load of route selection and routing data streams. For example, static route can be used in route to STUB network. 2) For route backup, configure static route in the backup line, with a lower priority than the main line.

Static route and dynamic route can coexist; layer3 switch will choose the route with the highest priority according to the priority of routing protocols. At the same time, static route can be introduced (redistribute) in dynamic route, and change the priority of the static route introduced as required.

2.2 Introduction to Default Route

Default route is a kind of static route, which is used only when no matching route is found. In the route table, default route in is indicated by a destination address of 0.0.0.0 and a network mask of 0.0.0.0, too. If the route table does not have the destination of a packet and has no default route configured, the packet will be discarded, and an ICMP packet will be sent to the source address indicate the destination address or network is unreachable.

2.3 Static Route Configuration Task List

1. Static route configuration

1. Static route configuration

Command	Explanation

Global mode	
ip route {*<ip-prefix>* *<mask>* | *<ip-prefix>*/*<prefix-length>*} {*<gateway-address>* | *<gateway-interface>*} [*<distance>*] **no ip route {*<ip-prefix>* *<mask>* | *<ip-prefix>*/*<prefix-length>*} [*<gateway-address>* | *<gateway-interface>*] [*<distance>*]**	Set static routing; the **no ip route {*<ip-prefix>* *<mask>* | *<ip-prefix>*/*<prefix-length>*} [*<gateway-address>* | *<gateway-interface>*] [*<distance>*]** command deletes a static route entry

2.4 Static Route Configuration Examples

The figure shown below is a simple network consisting of three layer3 switches, the network mask for all switches and PC is 255.255.255.0. PC-A and PC-C are connected via the static route set in SwtichA and SwitchC; PC3 and PC-B are connected via the static route set in SwtichC to SwitchB; PC-B and PC-C is connected via the default route set in SwitchB.

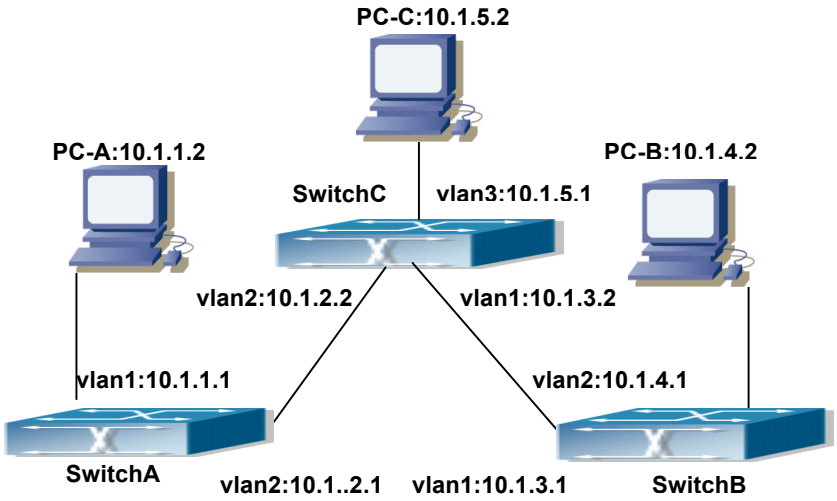


Fig 2-1 Static Route Configurations

Configuration steps:

Configuration of layer3 SwitchA

Switch#config

Switch (config) #ip route 10.1.5.0 255.255.255.0 10.1.2.2

Configuration of layer3 SwitchC

Switch#config

Next hop use the partner IP address

Switch(config)#ip route 10.1.1.0 255.255.255.0 10.1.2.1

Next hop use the partner IP address

Switch(config)#ip route 10.1.4.0 255.255.255.0 10.1.3.1

Configuration of layer3 SwitchB

Switch#config

Switch(config)#ip route 0.0.0.0 0.0.0.0 10.1.3.2

In this way, ping connectivity can be established between PC-A and PC-C, and PC-B and PC-C.

Chapter 3 RIP

3.1 Introduction to RIP

RIP is first introduced in ARPANET, this is a protocol dedicated to small, simple networks. RIP is a distance vector routing protocol based on the Bellman-Ford algorithm. Network devices running vector routing protocol send two kind of information to the neighboring devices regularly:

• Number of hops to reach the destination network, or metrics to use or number of networks to pass.

• What is the next hop, or the director (vector) to use to reach the destination network.

The distance vector Layer 3 switch send all their route selecting tables to the neighbor layer3 switches at regular interval. A layer3 switch will build their own route selecting information table based on the information received from the neighbor layer3 switches. Then, it will send this information to its own neighbor layer3 switches. As a result, the route selection table is built on second hand information, route beyond 15 hops will be deemed as unreachable.

RIP protocol is an optional routing protocol based on UDP. Hosts using RIP send and receive packets on UDP port 520. All layer3 switches running RIP send their route table to all neighbor layer3 switches every 30 seconds for update. If no information from the partner is received in 180 seconds, then the device is deemed to have failed and the network connected to that device is considered to be unreachable. However, the route of that layer3 switch will be kept in the route table for another 120 seconds before deletion.

As layer3 switches running RIP built route table with second hand information, infinite count may occur. For a network running RIP routing protocol, when an RIP route becomes unreachable, the neighboring RIP layer3 switch will not send route update packets at once, instead, it waits until the update interval timeout (every 30 seconds) and sends the update packets containing that route. If before it receives the updated packet, its neighbors send packets containing the information about the failed neighbor, “infinite count” will be resulted. In other words, the route of unreachable layer3 switch will be selected with the metrics increasing progressively. This greatly affects the route selection and route aggregation time.

To prevent “infinite count”, RIP provides mechanism such as “split horizon” and “triggered update” to solve route loop. “Split horizon” is done by avoiding sending to a gateway routes leaned from that gateway. There are two split horizon methods: “simple split horizon” and “poison reverse split horizon”. Simple split horizon deletes from the

route to be sent to the neighbor gateways the routes learnt from the neighbor gateways; poison reverse split horizon not only deletes the abovementioned routes, but set the costs of those routes to infinite. “Triggering update” mechanism defines whenever route metric changed by the gateway, the gateway advertise the update packets immediately, regardless of the 30 second update timer status.

There two versions of RIP, version 1 and version 2. RFC1058 introduces RIP-I protocol, RFC2453 introduces RIP-II, which is compatible with RFC1723 and RFC1388. RIP-I updates packets by packets broadcast, subnet mask and authentication is not supported. Some fields in the RIP-I packets are not used and are required to be all 0’s; for this reason, such all 0's fields should be checked when using RIP-I, the RIP-I packets should be discarded if such fields are non-zero. RIP-II is a more improved version than RIP-I. RIP-II sends route update packets by multicast packets (multicast address is 224.0.0.9). Subnet mask field and RIP authentication filed (simple plaintext password and MD5 password authentication are supported), and support variable length subnet mask. RIP-II used some of the zero field of RIP-I and require no zero field verification. switch send RIP-II packets in multicast by default, both RIP-I and RIP-II packets will be accepted.

Each layer3 switch running RIP has a route database, which contains all route entries for reachable destination, and route table is built based on this database. When a RIP layer3 switch sent route update packets to its neighbor devices, the complete route table is included in the packets. Therefore, in a large network, routing data to be transferred and processed for each layer3 switch is quite large, causing degraded network performance.

Besides the above mentioned, RIP protocol allows route information discovered by the other routing protocols to be introduced to the route table. It can also be as the protocol exchanging route messages with CE on PE routers, and supports the VPN route/transmitting examples.

The operation of RIP protocol is shown below:

1. Enable RIP. The switch sends request packets to the neighbor layer3 switches by broadcasting; on receiving the request, the neighbor devices reply with the packets containing their local routing information.
2. The Layer3 switch modifies its local route table on receiving the reply packets and sends triggered update packets to the neighbor devices to advertise route update information. On receiving the triggered update packet, the neighbor lay3 switches send triggered update packets to their neighbor lay3 switches. After a sequence of triggered update packet broadcast, all layer3 switches get and maintain the latest route information.

In addition, RIP layer3 switches will advertise its local route table to their neighbor devices every 30 seconds. On receiving the packets, neighbor devices maintain their local route table, select the best route and advertise the updated information to their own

neighbor devices, so that the updated routes are globally valid. Moreover, RIP uses a timeout mechanism for outdated route, that is, if a switch does not receive regular update packets from a neighbor within a certain interval (invalid timer interval), it considers the route from that neighbor invalid, after holding the route fro a certain interval (holddown timer interval), it will delete that route.

3.2 RIP Configuration Task List

1. Enable RIP (required)
   (1) Enable/disable RIP module.
   (2) Enable interface to send/receive RIP packets
2. Configure RIP protocol parameters (optional)
   (1) Configure RIP sending mechanism
      1) Configure specified RIP packets transmission address
      2) Configure RIP interface broadcast
   (2) Configure the RIP routing parameters
      1) Configure route introduction (default route metric, configure routes of the other protocols to be introduced in RIP)
      2) Configure interface authentication mode and password
      3) Configure the route deviation
      4) Configure and apply route filter
      5) Configure Split Horizon
   (3) Configure other RIP protocol parameters
      1) Configure the managing distance of RIP route
      2) Configure the RIP route capacity limit in route table
      3) Configure the RIP update, timeout, holddown and other timer.
      4) Configure the receiving buffer size of RIP UDP
3. Configure RIP-I/RIP-II switch
   (1) Configure the RIP version to be used in all interfaces
   (2) Configure the RIP version to send/receive in all interfaces
   (3) Configure whether to enable RIP packets sending/receiving for interfaces
4. Delete the specified route in RIP route table
5. Configure the RIP routing aggregation
   (1) Configure aggregation route of IPv4 route mode
   (2) Configure aggregation route of IPv4 interface configuration mode
   (3) Display IPv4 aggregation route information
6. Configure redistribution of OSPF routing to RIP

(1) Enable Redistribution of OSPF routing to RIP

(2) Display and debug the information about configuration of redistribution of OSPF routing to RIP

1. Enable RIP protocol

Applying RIP route protocol with basic configuration in switch is simple. Normally you only have to open the RIP switch and configure the segments running RIP, namely send and receive the RIP data packet by default RIP configuration. The version of data packet sending and receiving is variable when needed, allow/deny sending, receiving RIP data packet. Refer to 3.

Command	Explanation
Global Mode	
router rip **no router rip**	Enables RIP; the **no router rip** command disables RIP.
Router and address family configuration mode	
network <*A.B.C.D/M* | *ifname*|*vlan*> **no network <*A.B.C.D/M* | *ifname*|*vlan*>**	Enables the segment running RIP protocol; the **no network <*A.B.C.D/M* | *ifname*|*vlan*>** command deletes the segment.

2. Configure RIP protocol parameters

（1）Configure RIP packet transmitting mechanism

1）Configure the RIP data packet point-transmitting

2）Configure the Rip broadcast

Command	Explanation
Router Configuration Mode	
neighbor <*A.B.C.D*> **no neighbor <*A.B.C.D*>**	Specify the IP address of the neighbor router needs point-transmitting; the **no neighbor <*A.B.C.D*>** command cancels the appointed router.
passive-interface<*ifname*|*vlan*> **no passive-interface<*ifname*|*vlan* >**	Block the RIP broadcast on specified pot and the RIP data packet is only transmittable among Layer 3 switch configured with neighbor. The **no passive-interface<*ifname*|*vlan* >** command cancels the function.

（2）Configure RIP route parameters

1）Configure route introduction (default route metric, configure routes of the other protocols to be introduced in RIP)

Command	Explanation
Router Configuration Mode	
default-metric <*value*> **no default-metric**	Sets the default route metric for route to be introduced; the **no default-metric** command restores the default setting.
redistribute {kernel |connected| static| ospf | isis| bgp} [metric<*value*>] [route-map<*word*>] **no redistribute {kernel |connected| static| ospf | isis| bgp} [metric<*value*>] [route-map<*word*>]**	Redistribute the routes distributed in other routing protocols into the RIP data packet; the **no redistribute {kernel |connected| static| ospf | isis| bgp} [metric<value>] [route-map<word>]** command cancels the distributed route of corresponding protocols.
default-information originate **no default-information originate**	Generate a default route to the RIP protocol; the **no default-information originate** command cancels the feature.

2）Configure interface authentication mode and password

Command	Explanation
Interface configuration mode	
ip rip authentication mode { text| md5} **no ip rip authentication mode [text| md5]**	Sets the authentication method; the **no ip rip authentication mode [text| md5]** command cancels the authentication action.
ip rip authentication string <*text*> **no ip rip authentication string**	Sets the authentication key; the **no ip rip authentication string** command means no key is needed.
ip rip authentication key-chain <*name-of-chain*> **no ip rip authentication key-chain [<*name-of-chain*>]**	Sets the key chain used in authentication, the **no ip rip authentication key-chain [<*name-of-chain*>]** command means the key chain is not used.
ip rip authentication cisco-compatible **no ip rip authentication cisco-compatible**	After configure this command, configure MD5 authentication, then can receive RIP packet of Cisco, the no command restores the default configuration.
Global mode	

key chain <*name-of-chain*> **no key chain < *name-of-chain* >**	Enter keychain mode, and configure a key chain, the **no key chain < *name-of-chain* >** command deletes the key chain.
Keychain mode	
key <*keyid*> **no key <*keyid*>**	Enter the keychain-key mode and configure a key of the keychain; the **no key <*keyid*>** command deletes one key.
Keychain-key mode	
key-string <*text*> **no key-string <*text*>**	Configure the password used by the key, the **no key-string <*text*>** command deletes the password.
accept-lifetime <*start-time*> {<*end-time*>| duration<*seconds*>| infinite} **no accept-lifetime**	Configure a key on the key chain and accept it as an authorized time; the **no accept-lifetime** command deletes it.
send-lifetime <*start-time*> {<*end-time*>| duration<*seconds*>| infinite} **no send-lifetime**	Configure the transmitting period of a key on the key chain; the **no send-lifetime** command deletes the send-lifetime.

3）Configure the route deviation

Command	Explanation
Router configuration mode	
offset-list <*access-list-number* | *access-list-name*> {in | out } <*number*> [<*ifname*>] **no offset-list <*access-list-number* |*access-list-name*> {in|out }<*number* >[<*ifname*>]**	Configure that provide a deviation value to the route metric value when the port sends or receives RIP data packet; the **no offset-list <access-list-number |access-list-name> {in|out } <number >[<ifname>]** command removes the deviation table.

4）Configure and apply the route filtering

Command	Explanation
Router configuration mode	

distribute-list {< *access-list-number* |*access-list-name* >|prefix<*prefix-list-name*>}{in|out} [<*ifname*>] **no distribute-list {< *access-list-number* |*access-list-name* >|prefix<*prefix-list-name*>}{in|out} [<*ifname*>]**	Configure and apply the access table and prefix table to filter the routes. The **no distribute-list {< *access-list-number* |*access-list-name*>|prefix<*prefix-list-name*>}{in|out} [<*ifname*>]** command means do not use the access table and prefix table.

5）Configure the split horizon

Command	Explanation
Interface configuration mode	
ip rip split-horizon [poisoned] **no ip rip split-horizon**	Configure that take the split horizon when the port sends data packets; poisoned for poison reverse the **no ip rip split-horizon** command cancels the split horizon.

（3）Configure other RIP protocol parameters

1）Configure RIP routing priority

2）Configure the RIP route capacity limit in route table

3）Configure timer for RIP update, timeout and hold-down

4）Configure RIP UDP receiving buffer size

Command	Explanation
Router configuration mode	
distance <*number*> [<*A.B.C.D/M*>] [<*access-list-name*|*access-list-number* >] **no distance [<*A.B.C.D/M*>]**	Specify the route administratively distance of RIP protocol; the **no distance [<*A.B.C.D/M*>]** command restores the default value 120.
maximum-prefix <*maximum-prefix*>[<*threshold*>] **no maximum-prefix <*maximum-prefix* >** **no maximum-prefix**	Configure the maximum of RIP route; the **no maximum-prefix <*maximum-prefix* >** **no maximum-prefix** command cancels the limit.
timers basic <*update*> <*invalid*> <*garbage*> **no timers basic**	Adjust the update, timeout and garbage collection time, the **no timers basic** command restores the default configuration.
recv-buffer-size <*size*> **no recv-buffer-size**	The command configures the UDP receiving buffer size of the RIP; the **no recv-buffer-size** command restores the system default values.

3. Configure RIP-I/RIP-II toggling

（1）Configure the RIP version to be used in all ports

Command	Explanation
RIP configuration mode	
version { 1 | 2 } **no version**	Configure the versions of all the RIP data packets transmitted/received by the Layer 3 switch port sending/receiving the **no version** command restores the default configuration, version 2.

（**2**）Configure the RIP version to send/receive in all ports.

（**3**）Configure whether to enable RIP packets sending/receiving for ports

Command	Explanation
Interface configuration mode	
ip rip send version { 1 | 1-compatible | 2 } **no ip rip send version**	Sets the version of RIP packets to send on all ports; the **no ip rip send version** command set the version to the one configured by the version command.
ip rip receive version {1 | 2 | } **no ip rip receive version**	Sets the version of RIP packets to receive on all ports; the no action of this command set the version to the one configured by the version command.
ip rip receive-packet **no ip rip receive-packet**	Enables receiving RIP packets on the interface; the **no ip rip receive-packet** command close data receiving on this port.
ip rip send-packet **no ip rip send-packet**	Enables sending RIP packets on the interface; the **no ip rip send-packet** command disables sending RIP packets on the interface.

4. Delete the specified route in RIP route table

Command	Explanation
Admin Mode	
clear ip rip route {<*A.B.C.D/M*>|kernel|static|connected |rip|ospf|isis|bgp|all}	The command deletes a specified route from the RIP route table.

5. Configure the RIP routing aggregation

（1） Configure IPv4 aggregation route globally

Command	Explanation

Router Configuration Mode	
ip rip aggregate-address A.B.C.D/M **no ip rip aggregate-address A.B.C.D/M**	To configure or delete IPv4 aggregation route globally.

（2） Configure IPv4 aggregation route on interface

Command	Explanation
Interface Configuration Mode	
ip rip aggregate-address A.B.C.D/M **no ip rip aggregate-address A.B.C.D/M**	To configure or delete IPv4 aggregation route on interface.

（3） Display IPv4 aggregation route information

Command	Explanation
Admin Mode and Configuration Mode	
show ip rip aggregate	To display aggregation route information.

6. Configure redistribution of OSPF routing to RIP

(1) Enable Redistribution of OSPF routing to RIP

Command	Explanation
Router RIP Configuration Mode	
redistribute ospf [<*process-id*>] [metric <*value*>] [route-map <*word*>] **no redistribute ospf [<*process-id*>]**	To enable or disable the redistribution of OSPF routing to RIP.

(2) Display and debug the information about configuration of redistribution of OSPF routing to RIP

Command	Explanation
Admin Mode and Configuration Mode	
show ip rip redistribute	To display the information about configuration of redistribute from other routing.
Admin Mode	
debug rip redistribute message send **no debug rip redistribute message send** **debug rip redistribute route receive** **no debug rip redistribute route receive**	To enable or disable debugging messages sent by RIP for redistribution of OSPF routing. To enable or disable debugging messages received from NSM.

3.3 RIP Examples

3.3.1 Typical RIP Examples

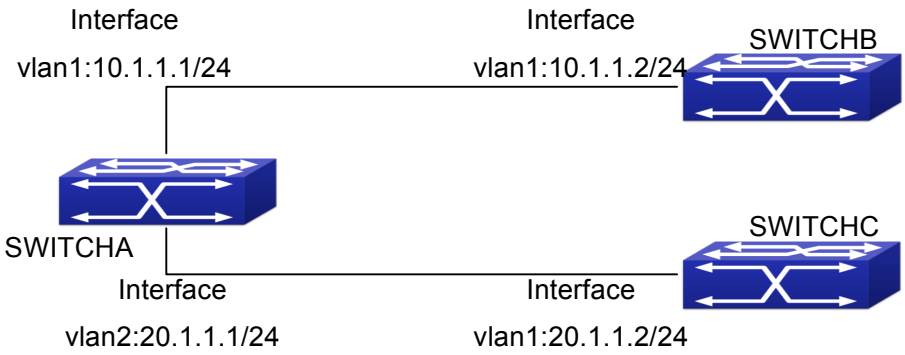


Fig 3-1 RIP example

In the figure shown above, a network consists of three Layer 3 switches, in which SwitchA connected with SwitchB and SwitchC, and RIP routing protocol is running in all of the three switches. SwitchA (interface vlan1： 10.1.1.1,interface vlan2： 20.1.1.1) exchanges Layer 3 switch update messages only with SwitchB (interface vlan1：10.1.1.2), but not with SwitchC (interface vlan 2: 20.1.1.2).

SwitchA, SwitchB, SwitchC configurations are as follows:

a) Layer 3 SwitchA：

Configure the IP address of interface vlan 1

SwitchA#config

SwitchA(config)# interface vlan 1

SwitchA(Config-if-Vlan1)# ip address 10.1.1.1 255.255.255.0

SwitchA(config-if-Vlan1)#

Configure the IP address of interface vlan 2

SwitchA(config)# vlan 2

SwitchA(Config-Vlan2)# switchport interface ethernet 1/0/2

Set the port Ethernet1/0/2 access vlan 2 successfully

SwitchA(Config-Vlan2)# exit

SwitchA(config)# interface vlan 2

SwitchA(Config-if-Vlan2)# ip address 20.1.1.1 255.255.255.0

Initiate RIP protocol and configure the RIP segments

SwitchA(config)#router rip

SwitchA(config-router)#network vlan 1

SwitchA(config-router)#network vlan 2

SwitchA(config-router)#exit

Configure that the interface vlan 2 do not transmit RIP messages to SwitchC

SwitchA(config)#router rip

SwitchA(config-router)#passive-interface vlan 2

SwitchA(config-router)#exit

SwitchA(config) #

b) Layer 3 SwitchB

Configure the IP address of interface vlan 1

SwitchB#config

SwitchB(config)# interface vlan 1

SwitchB(Config-if-Vlan1)# ip address 10.1.1.2 255.255.255.0

SwitchB(Config-if-Vlan1)exit

Initiate RIP protocol and configure the RIP segments

SwitchB(config)#router rip

SwitchB(config-router)#network vlan 1

SwitchB(config-router)#exit

c) Layer 3 SwitchC

SwitchC#config

SwitchC(config)# interface vlan 1

Configure the IP address of interface vlan 1

SwitchC(Config-if-Vlan1)# ip address 20.1.1.2 255.255.255.0

SwitchC(Config-if-Vlan1)#exit

Initiate RIP protocol and configure the RIP segments

SwitchC(config)#router rip

SwitchC(config-router)#network vlan 1

SwitchC(config-router)#exit

3.3.2 Typical Examples of RIP aggregation function

The application topology as follows：

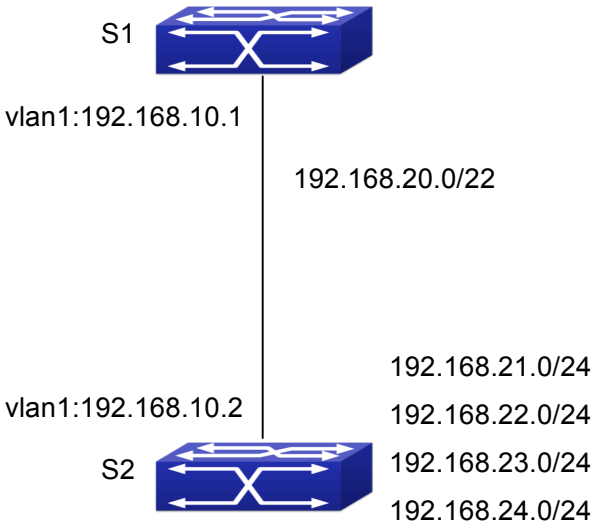


Fig 3-2 Typical application of RIP aggregation

As the above network topology, S2 is connected to S1 through interface vlan1, there are other 4 subnet routers of S2, which are 192.168.21.0/24, 192.168.22.0/24, 192.168.23.0/24, 192.168.24.0/24. S2 supports route aggregation, and to configure aggregation route 192.168.20.0/22 in interface vlan1 of S2, after that, sending router messages to S1 through vlan1, and put the four subnet routers aggregated to one router as 192.168.20.0/22, and send to S1, and not send subnet to neighbor. It can reduce the router table of S1, save the memory.

S1 configuration list:

S1(config)#router rip

S1(config-router) #network vlan 1

S2 configuration list:

S2(config)#router rip

S2(config-router) #network vlan 1

S2(config-router) #exit

S2(config)#in vlan 1

S2(Config-if-Vlan1)# ip rip agg 192.168.20.0/22

3.4 RIP Troubleshooting

The RIP protocol may not be working properly due to errors such as physical connection, configuration error when configuring and using the RIP protocol. So users should pay attention to following:

- First ensure the physic connection is correct
- Second, ensure the interface and chain protocol are UP (use **show interface** command)

- Then initiate the RIP protocol (use **router rip** command) and configure the segment (use **network** command) and set RIP protocol parameter on corresponding interfaces, such as the option between RIP-I and RIP-II
- After that, one feature of RIP protocol should be noticed ---the Layer 3 switch running RIP protocol sending route updating messages to all neighboring Layer 3 switches every 30 seconds. A Layer 3 switch is considered inaccessible if no route updating messages from the switch is received within 180 seconds, then the route to the switch will remains in the route table for 120 seconds before it is deleted. Therefore, if to delete a RIP route, this route item is assured to be deleted from route table after 300 seconds.
- When exchanging routing messages with CE using RIP protocol on the PE router, we should first create corresponding VPN routing/transmitting examples to associate with corresponding interfaces. Then enter the RIP address family mode configuring corresponding parameters. If the RIP routing problem remains unresolved, please use debug rip command to record the debug message in three minutes, and send them to our technical service center.

Chapter 4 RIPng

4.1 Introduction to RIPng

RIPng is first introduced in ARPANET, this is a protocol dedicated to small, simple networks. RIPng is a distance vector routing protocol based on the Bellman-Ford algorithm. Network devices running vector routing protocol send 2 kind of information to the neighboring devices regularly:

• Number of hops to reach the destination network, or metrics to use or number of networks to pass.

• What is the next hop, or the director (vector) to use to reach the destination network.

Distance vector layer3 switches send all their route selecting tables to the neighbor layer3 switches at regular interval. A layer3 switch will build their own route selecting information table based on the information received from the neighbor layer3 switches. Then, it will send this information to its own neighbor layer3 switches. As a result, the route selection table is built on second hand information, route beyond 15 hops will be deemed as unreachable.

RIPng is an optional routing protocol based on UDP. Hosts using RIPng send and receive packets on UDP port 521. All layer3 switches running RIP send their route table to all neighbor layer3 switches every 30 seconds for update. If no information from the partner is received in 180 seconds, then the device is deemed to have failed and the network connected to that device is considered to be unreachable. However, the route of that layer3 switch will be kept in the route table for another 120 seconds before deletion.

As layer3 switches running RIPng build route table with second hand information, infinite count may occur. For a network running RIPng routing protocol, when a RIPng route becomes unreachable, the neighboring RIPng layer3 switch will not send route update packets at once, instead, it waits until the update interval timeout (every 30 seconds) and sends the update packets containing that route. If before it receives the updated packet, its neighbors send packets containing the information about the failed neighbor, “infinite count” will be resulted. In other words, the route of unreachable layer3 switch will be selected with the metrics increasing progressively. This greatly affects the route selection and route aggregation time.

To avoid “infinite count”, RIPng provides mechanism such as “split horizon” and “triggered update” to solve route loop. “Split horizon” is done by avoiding sending to a gateway routes leaned from that gateway. There are two split horizon methods: “simple split horizon” and “poison reverse split horizon”. Simple split horizon deletes from the

route to be sent to the neighbor gateways the routes learnt from the neighbor gateways; poison reverse split horizon not only deletes the abovementioned routes, but set the costs of those routes to infinite. “Triggering update” mechanism defines whenever route metric changed by the gateway, the gateway advertise the update packets immediately other than wait for the 30 sec timer.

So far the RIPng protocol has got only one version----Version1: RIPng protocol is introduced in RFC 2080. RIPng transmits updating data packet by multicast data packet (multicast address FF02::9)

Each layer3 switch running RIPng has a route database, which contains all route entries for reachable destination, and route table is built based on this database. When a RIPng layer3 switch sent route update packets to its neighbor devices, the complete route table is included in the packets. Therefore, in a large network, routing data to be transferred and processed for each layer3 switch is quite large, causing degraded network performance.

Besides the above mentioned, RIPng protocol allows IPv6 route information discovered by the other routing protocols to be introduced to the route table.

The operation of RIPng protocol is shown below:

1. Enable RIPng The switch sends request packets to the neighbor layer3 switches by broadcasting; on receiving the request, the neighbor devices reply with the packets containing their local routing information.
2. The Layer3 switch modifies its local route table on receiving the reply packets and sends triggered update packets to the neighbor devices to advertise route update information. On receiving the triggered update packet, the neighbor lay3 switches send triggered update packets to their neighbor lay3 switches. After a sequence of triggered update packet broadcast, all layer3 switches get and maintain the latest route information.

In addition, RIPng layer3 switches will advertise its local route table to their neighbor devices every 30 seconds. On receiving the packets, neighbor devices maintain their local route table, select the best route and advertise the updated information to their own neighbor devices, so that the updated routes are globally valid. Moreover, RIP uses a timeout mechanism for outdated route, that is, if a switch does not receive regular update packets from a neighbor within a certain interval (invalid timer interval), it considers the route from that neighbor invalid, after holding the route fro a certain interval (garbage collect timer interval), it will delete that route.

As a result of continuous development of IPv6 network, it has the network environment of nonsupport IPv6 sometimes, so it needs to do the IPv6 operation by tunnel. Therefore, our RIPng supports configuration on configure tunnel, and passes through nonsupport IPv6 network by unicast packet of IPv4 encapsulation.

4.2 RIPng Configuration Task List

RIPng Configuration Task List:

1. Enable RIPng protocol (required)
   （1） Enable/disable RIPng protocol
   （2） Configure the interfaces running RIPng protocol
2. Configure RIPng protocol parameters (optional)
   （1） Configure RIPng sending mechanism
      1) Configure specified RIPng packets transmission address
   （2） Configure RIP routing parameters
      1) Configure route introduction (default route metric, configure routes of the other protocols to be introduced in RIPng)
      2) Configure the route deviation
      3) Configure and apply route filter
      4) Configure split horizon
3. Configure other RIPng parameters
   (1) Configure timer for RIPng update, timeout and hold-down
4. Delete the specified route in RIPng route table
5. Configure RIPng route aggregation
   （1） Configure aggregation route of IPv6 route mode
   （2） Configure aggregation route of IPv6 interface configuration mode
   （3） Display IPv6 aggregation route information
6. Configure redistribution of OSPFv3 routing to RIPng
   （1） Enable redistribution of OSPFv3 routing to RIPng
   （2） Display and debug the information about configuration of redistribution of OSPFv3 routing to RIPng

1. Enable RIPng protocol

Applying RIPng route protocol with basic configuration in switch is simple. Normally you only have to open the RIPng switch and configure the segments running RIPng, namely send and receive the RIPng data packet by default RIPng configuration.

Command	Explanation
Global mode	
[no] router IPv6 rip	Enables the RIPng protocol; the **no router IPv6 rip** command shuts the RIPng protocol.
Interface configuration mode	

[no] IPv6 router rip	Configure the interface to run RIPng protocol; the **no IPv6 router rip** command set the interface not run RIPng protocol.

2. Configure RIPng protocol parameters

（1）Configure RIPng sending mechanism

1）Configure the RIPng data packets point-transmitting

Command	Explanation
Router configuration mode	
[no] neighbor *<IPv6-address>* *<ifname>*	Specify the IPv6 Link-local address and interface of the neighboring route needs point-transmitting; the **no neighbor *<IPv6-address>* *<ifname>*** command cancels the appointed router.
[no] passive-interface *<ifname>*	Block the RIPng multicast on specified port and the RIPng data packet is only transmittable among Layer 3 switch configured with neighbor. The **no passive-interface *<ifname>*** command cancels the function.

（2）Configure RIP routing parameters

1）Configure route introduction (default route metric, configure routes of the other protocols to be introduced in RIP)

Command	Explanation
Router configuration mode	
default-metric *<value>* **no default-metric**	Configure the default metric of distributed route; the **no default-metric** command restores the default configuration 1.
[no]redistribute {kernel |connected| static| ospf| isis| bgp} [metric<*value*>] [route-map<*word*>]	Redistribute the routes distributed in other route protocols into the RIPng data packet; the **no redistribute {kernel |connected| static| ospf| isis| bgp} [metric<*value*>] [route-map<*word*>]** command cancels the distributed route of corresponding protocols.
[no]default-information originate	Generate a default route to the RIPng protocol; the **no default-information originate** command cancels the feature.

2）Configure the route offset

Command	Explanation
Router configuration mode	
[no] offset-list <*access-list-number*|*access-list-name*> {in|out} <*number* > [<*ifname*>]	Configure that provide a deviation value to the route metric value when the port sends or receives RIPng data packet; the **no offset-list <*access-list-number* |*access-list-name*> {in|out} <*number* > [<*ifname*>]** command removes the deviation table.

3）Configure and apply route filter and route aggregation

Command	Explanation
Router configuration mode	
[no] distribute-list {<*access-list-number*|*access-list-name*> | prefix<*prefix-list-name*>} {in|out} [<*ifname*>]	Set to filter the route when the interface sends and receives RIPng data packets. The **no distribute-list {< *access-list-number* |*access-list-name* > | prefix<*prefix-list-name*>} {in|out} [<*ifname*>]** command means do not set the route filter.
[no]aggregate-address <*IPv6-address*>	Configure route aggregation, the **no aggregate-address <*IPv6-address*** command cancels the route aggregation.

4）Configure split horizon

Command	Explanation
Interface configuration mode	
IPv6 rip split-horizon [poisoned]	Configure that take the split-horizon when the port sends data packets, **poisoned** means with poison reverse.
no IPv6 rip split-horizon	Cancel the split-horizon.

3. Configure other RIPng protocol parameters

(1) Configure timer for RIPng update, timeout and hold-down

Command	Explanation
Router configuration mode	

timers basic <*update*> <*invalid*> <*garbage*> **no timers basic**	Adjust update, timeout and garbage recycle of RIPng timer, the **no timers basic** command restores the default configuration.

4. Delete the specified route in RIPng route table

Command	**Explanation**
Admin Mode	
clear IPv6 rip route {<*IPv6-address*>|kernel|static|connected|rip|ospf|isis|bgp|all}	The command deletes a specified route from the RIP route table.

5. Configure RIPng route aggregation

(1) Configure IPv6 aggregation route globally

Command	Explanation
Router Configuration Mode	
ipv6 rip aggregate-address X:X::X:X/M **no ipv6 rip aggregate-address X:X::X:X/M**	To configure or delete IPv6 aggregation route globally.

(2) Configure IPv6 aggregation route on interface

Command	Explanation
Interface Configuration Mode	
ipv6 rip aggregate-address X:X::X:X/M **no ipv6 rip aggregate-address X:X::X:X/M**	To configure or delete IPv6 aggregation route on interface.

(3) Display IPv6 aggregation route information

Command	Explanation
Admin Mode and Configuration Mode	
show ipv6 rip aggregate	To display IPv6 aggregation route information, such as aggregation interface, metric, numbers of aggregation route, times of aggregation.

6. Configure redistribution of OSPFv3 routing to RIPng

(1)Enable redistribution of OSPFv3 routing to RIPng

Command	Explanation

Router IPv6 RIP Configuration Mode	
redistribute ospf [<*process-tag*>] [metric<*value*>] [route-map<*word*>] **no redistribute ospf [<*process-tag*>]**	To enable or disable redistribution of OSPFv3 routing for RIPng.

(2) Display and debug the information about configuration of redistribution of OSPFv3 routing to RIPng

Command	Explanation
Admin Configuration Mode	
show ipv6 rip redistribute	To display RIPng routing which is redistributed from other routing protocols.
Admin Mode	
debug ipv6 rip redistribute message send **no debug ipv6 rip redistribute message send** **debug ipv6 rip redistribute route receive** **no debug ipv6 rip redistribute route receive**	To enable or disable debugging messages sent by RIPng for redistribution of OSPFv3 routing. To enable or disable debugging route messages received from NSM.

4.3 RIPng Configuration Examples

4.3.1 Typical RIPng Examples

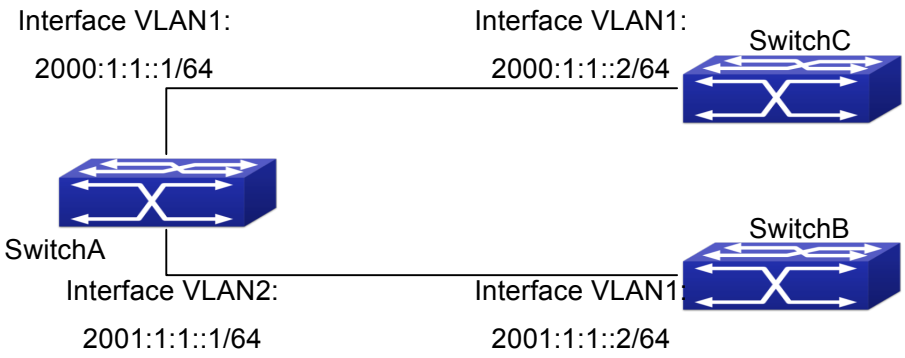


Fig 4-1 RIPng Example

As shown in the above figure, a network consists of three layer 3 switches. SwitchA and SwitchB connect to SwitchC through interface vlan1 and vlan2. All the three switches are running RIPng. Assume SwitchA (VLAN1：2001:1:1::1/64 and VLAN2：2001:1:1::1/64)

exchange update information with SwitchB (VLAN1： 2001:1:1::2/64) only, update information is not exchanged between SwitchA and SwitchC (VLAN1：2001:1:1::2/64).

The configuration for SwitchA, SwitchB and SwitchC is shown below:

Layer 3 SwitchA

Enable RIPng protocol

SwitchA(config)#router IPv6 rip

SwitchA(config-router)#exit

Configure the IPv6 address in vlan1 and configure vlan1 to run RIPng

SwitchA#config

SwitchA(config)# interface Vlan1

SwitchA(config-if-Vlan1)# IPv6 address 2000:1:1::1/64

SwitchA(config-if-Vlan1)#IPv6 router rip

SwitchA(config-if-Vlan1)#exit

Configure the IPv6 address in vlan2 and configure vlan2 to run RIPng

SwitchA(config)# interface Vlan2

SwitchA(config-if-Vlan2)#IPv6 address 2001:1:1::1/64

SwitchA(config-if-Vlan2)#IPv6 router rip

SwitchA(config-if-Vlan2)#exit

Configure the interface vlan1 do not send RIPng messages to SwitchC

SwitchA(config)#

SwitchA(config-router)#passive-interface Vlan1

SwitchA(config-router)#exit

Layer 3 SwitchB

Enable RIPng protocol

SwitchB (config)#router IPv6 rip

SwitchB (config-router-rip)#exit

Configure the IPv6 address and interfaces of Ethernet port vlan1 to run RIPng

SwitchB#config

SwitchB(config)# interface Vlan1

SwitchB(config-if)# IPv6 address 2001:1:1::2/64

SwitchB(config-if)#IPv6 router rip

SwitchB(config-if)exit

Layer 3 SwitchC

Enable RIPng protocol

SwitchC(config)#router IPv6 rip

SwitchC(config-router-rip)#exit

Configure the IPv6 address and interfaces of Ethernet port vlan1 to run RIPng

SwitchC#config

SwitchC(config)# interface Vlan1

SwitchC(config-if)# IPv6 address 2000:1:1::2/64

SwitchC(config-if)#IPv6 router rip

SwitchC(config-if)exit

4.3.2 RIPng Aggregation Route Function Typical Examples

The application topology as follows:

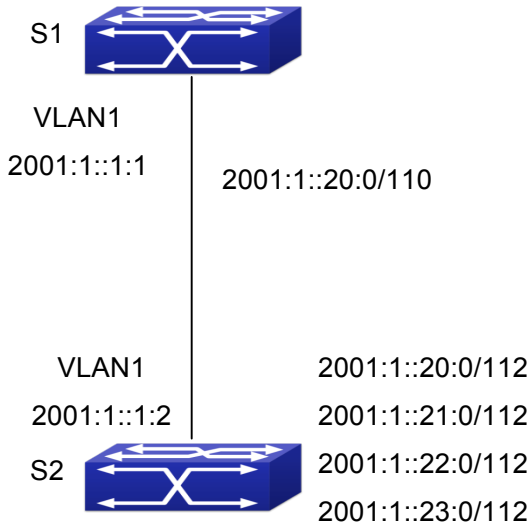


Fig 4-2 Typical application of RIPng aggregation

As the above network topology, S2 is connected to S1 through interface vlan1, there are other 4 subnet routers of S2, which are 2001:1::20:0/112, 2001:1::21:0/112, 2001:1::22:0/112, 2001:1::23:0/112. S2 supports route aggregation, and to configure aggregation route 2001:1::20:0/110 in interface vlan1 of S2, after that, sending router messages to S2 through vlan1, and put the four subnet routers aggregated to one router as 2001:1::20:0/110, and send to S1, and not send subnet to neighbor. It can reduce the router table of S1, save the memory.

S1 configuration list:

S1(config)#router ipv6 rip

S1(config-router) #network vlan 1

S2 configuration list:

S2(config)#router ipv6 rip

S2(config-router) #network vlan 1

S2(config-router) #exit

S2(config)#in vlan 1

S2(Config-if-Vlan1)# ipv6 rip agg 2001:1::20:0/110

4.4 RIPng Troubleshooting

The RIPng protocol may not be working properly due to errors such as physic connection, configuration error when configuring and using the RIPng protocol. So users should pay attention to the following:

- ☞ First ensure the physic connection is correct and the IP Forwarding command is open
- ☞ Second, ensure the interface and link layer protocol are UP (use **show interface** command)
- ☞ Then initiate the RIPng protocol (use **router IPv6 rip** command) and configure the port (use **IPv6 router** command), and set RIPng protocol parameter on corresponding interfaces.
- ☞ After that, a RIPng protocol feature should be noticed ---the Layer 3 switch running RIPng transmits the route updating messages every 30 seconds. A Layer 3 switch is considered inaccessible if no route updating messages from the switch are received within 180 seconds, then the route to the switch will remains in the route table for 120 seconds before it is deleted. Therefore, if to delete a RIPng route, this route item is assured to be deleted from route table after 300 seconds.
- ☞ If the RIP routing problem remains unresolved, please use **debug IPv6 rip** command to record the debug message in three minutes, and send them to our technical service center.

Chapter 5 Black Hole Routing Manual

5.1 Introduction to Black Hole Routing

Black Hole Routing is a special kind of static routing which drops all the datagrams that match the routing rule.

5.2 IPv4 Black Hole Routing Configuration Task

1. Configure IPv4 Black Hole Routing

1. Configure IPv4 Black Hole Routing

Command	Explaination
Global Configuration Mode	
ip route {*<ip-prefix>* *<mask>*|*<ip-prefix>*/*<prefix-length>*} null0 [*<distance>*] **no ip route {*<ip-prefix>* *<mask>*|*<ip-prefix>*/*<prefix-length>*} null0**	To configure the static Black Hole Routing. The no form of this command will remove the specified Black Hole Routing configuration.

5.3 IPv6 Black Hole Routing Configuration Task

1. Enable the IPv6 function
2. Configure the IPv6 Black Hole Routing

1. Enable the IPv6 function

Command	Explaination
Global Configuration Mode	
ipv6 enable	To enable the IPv6 function on the switch.

2. Configure IPv6 Black Hole Routing

Command	Explaination
Global Configuration Mode	

ipv6 route ***<ipv6-prefix/prefix-length>*** **null0 [<*precedence*>]** **no ipv6 route** ***<ipv6-prefix/prefix-length>*** **null0**	To configure static IPv6 Black Hole Routing. The no form of this command will remove the specified configuration.

5.4 Black Hole Routing Configuration Exmaples

Example 1: IPv4 Black Hole Routing function.

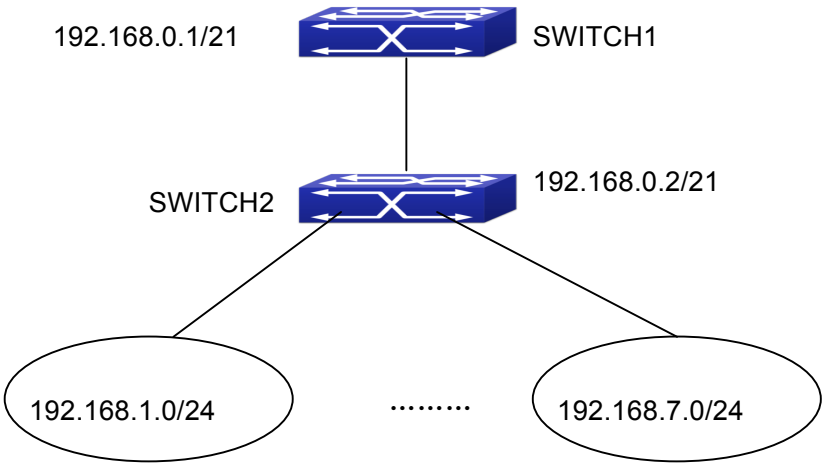


Fig 5-1 IPv4 Black Hole Routing Configuration Example

As it is shown in the figure, in Switch 2, eight in all interfaces are configured as Layer 3 VLAN interfaces for access interfaces. The network addresses are 192.168.1.0/24 ~ 192.268.7.0/24. A default routing is configured on Switch 2 to connect to Switch 1. And a backward default routing is configured on Switch 1 to Switch 2, whose network address is 192.168.0.0/21. Commonly, this configuration will work well. However, if one of the Layer 3 interfaces in Switch 2 goes down, for example, the interface belonged to 192.168.1.0/24. When datagrams arrives at VLAN1 in Switch 2, there will be no routing rules for these datagrams. The switch then will forward these datagrams according to the default routing, back to Switch 1. When Switch 1 receives these datagrams, it will forward them back to Switch 2. Thus, loopback exists. To solve this problem, Black Hole Routing can be introduced on Switch 2.

ip route 192.168.0.0/21 null0 50

Then Switch 2 will drop the datagrams from interface VLAN1 that match the Black

Hole Routing rule. And loopback routing is prevented.

Configuration steps are listed as below:

Switch#config

Switch(config)#ip route 192.168.0.0/21 null0 50

Example 2: IPv6 Black Hole Routing function.

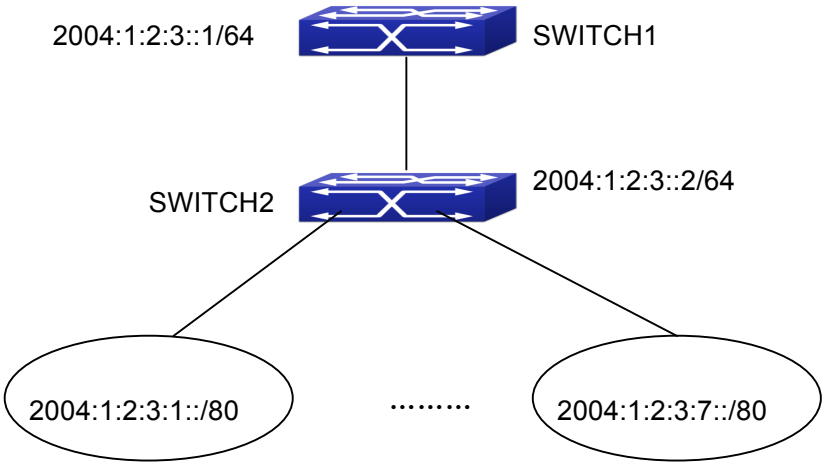


Fig 5-2 IPv6 Black Hole Routing Configuration Example

As it is shown in the figure, in Switch 2, eight in all interfaces are configured as Layer 3 VLAN interfaces for access interfaces. The network addresses are 2004:1:2:3:1/80~2004:1:2:3:7/80. A default routing is configured on Switch 2 to connect to Switch 1. And a backward default routing is configured on Switch 1 to Switch 2, whose network address is 2004:1:2:3::/64. Commonly, this configuration will work well. However, if one of the Layer 3 interfaces in Switch 2 goes down, for example, the interface belonged to 2004:1:2:3:1/80. When datagrams arrives at VLAN1 in Switch 2, there will be no routing rules for these datagrams. The switch then will forward these datagrams according to the default routing, back to Switch 1. When Switch 1 receives these datagrams, it will forward them back to Switch 2. Thus, loopback exists. To solve this problem, Black Hole Routing can be introduced on Switch 2.

ipv6 route 2004:1:2:3::/64 null0 50

Then Switch 2 will drop the datagrams from interface VLAN1 that match the Black Hole Routing rule. And loopback routing is prevented.

Configuration steps are listed as below:

Switch#config

Switch(config)#ipv6 route 2004:1:2:3::/64 null0 50

5.5 Black Hole Routing Troubleshooting

When configuring the Black Hole Routing function, the configuration may not work due to some reasons such as incorrect network address mask, and incorrect management distance. Attention should be paid to the following items:

- ☞ IPv6 should be enabled before IPv6 Black Hole Routing can work.
- ☞ It is suggested that the length of the network address mask should be longer than that of normal routing configuration, in order to prevent the Black Hole Routing from intervening other routing configuration.
- ☞ When the network address mask of Black Hole Routing configuration is the same with some other configuration, it is suggested that the distance of Black Hole Routing is set lower.

For problems that cannot be fixed through above methods, please issue the command show ip route distance and show ip route fib, and show l3. And copy and paste the output of the commands, and send to the technical service center of our company.

Chapter 6 BFD

6.1 Introduction to BFD

BFD (Bidirectional Forwarding Detection) provides a detection mechanism to quickly detect and monitor the connectivity of links in networks. To improve network performance, between protocol neighbors must quickly detect communication failures to restore communication through backup paths as soon as possible.

BFD provides a general-purpose, standard, medium-independent and protocol-independent fast failure detection mechanism. It can uniformly and quickly detect the failures of the bidirectional forwarding paths between two network devices for superstratum protocols, such as routing protocols and Multiprotocol Label Switching (MPLS). BFD establishes session between two network devices to monitor their bidirectional forwarding paths to serve for superstratum protocols. However, there is no discovery mechanism for BFD, it is notified by superstratum protocol to establish sessions. After a session is established, if no BFD control packet is received from the peer within detection time, it notifies the failure to superstratum protocol which will take appropriate measures.

6.2 BFD Configuration Task List

1. Configure BFD basic function
2. Configure BFD for RIP (ng)
3. Configure BFD for static route (IPv6)
4. Configure BFD for VRRP (v3)

1. Configure BFD basic function

Command	Explanation
Global Mode	
bfd mode{active | passive} **no bfd mode**	Configure the mode before establishing BFD session, the default is active mode. No command restores active mode.
bfd authentication key <*1-255*> text <*WORD*> **no bfd authentication key <*1-255*>**	Configure key and authentication character

	string encrypted with text for BFD, no command deletes the configured key.
bfd authentication key <*1-255*> md5 <*WORD*> **no bfd authentication key**	Configure key and authentication character string encrypted with md5 for BFD, no command deletes the configured key.
Interface Mode	
bfd interval <*value1*> min_rx <*value2*> multiplier <*value3*> **no bfd interval**	Configure the minimum transmission interval and the multiplier of session detection for BFD control packets, no command restores the default detection multiplier.
bfd min-echo-receive-interval <*value*> **no bfd min-echo-receive-interval**	Configure the minimum receiving interval for BFD control packets, no command restores its default value.
bfd echo **no bfd echo**	Enable bfd echo, no command disables the function.
bfd echo-source-ip <*ipv4-address*> **no bfd echo-source-ip**	Detect link fault by configuring source address of echo packets, no command deletes the configured source address of echo packets.
bfd echo-source-ipv6 <*ipv6-address*> **no bfd echo-source-ipv6**	Detect link fault by configuring source address of echo packets, no command deletes the configured source address of echo packets.
bfd authentication key <*1-255*> **no bfd authentication key**	Enable BFD authentication and configure key for interface, no command

	disables BFD authentication.

2. Configure BFD for RIP (ng)

Command	Explanation
Interface Mode	
rip bfd enable **no rip bfd enable**	Configure BFD for RIP protocol on the specific interface, no command disables BFD for RIP protocol.
ipv6 rip bfd enable **no ipv6 rip bfd enable**	Configure BFD for RIPng protocol on the specific interface, no command cancels the configuration.

3. Configure BFD for static route (IPv6)

Command	Explanation
Global Mode	
ip route {vrf <*name*> <*ipv4-address*> | <*ipv4-address*>} mask <*nexthop*> bfd **no ip route {vrf <*name*> <*ipv4-address*> | <*ipv4-address*>} mask <*nexthop*> bfd**	Configure BFD for the static route, no command cancels the configuration.
ipv6 route {vrf <*name*> <*ipv6-address*> | <*ipv6-address*>} prefix <*nexthop*> bfd **no ipv6 route {vrf <*name*> <*ipv6-address*> | <*ipv6-address*>} prefix <*nexthop*> bfd**	Configure BFD for the static IPv6 route, no command cancels the configuration.

4. Configure BFD for VRRP (v3)

Command	Explanation
VRRP(v3) Group Configuration Mode	
bfd enable **no bfd enable**	Enable BFD for VRRP(v3) protocol and enable BFD detection on this group, no command disables the function.

6.3 Examples of BFD

6.3.1 Example for Linkage of BFD and Static Route

Example:

Configure a static route to 14.1.1.0/24 on Switch A and configure a static route to 15.1.1.0/24 on Switch B. Both switches enable BFD detection. When the link between Switch A and Switch B is failing, BFD can detect it immediately.

Configuration procedure:

Switch A:

Switch#config

Switch(config)#interface vlan 12

Switch(config-if-vlan12)#ip address 12.1.1.1 255.255.255.0

Switch(config)#interface vlan 15

Switch(config-if-vlan15)#ip address 15.1.1.1 255.255.255.0

Switch(config)#ip route 14.1.1.0 255.255.255.0 12.1.1.2 bfd

Switch B:

Switch#config

Switch(config)#interface vlan 12

Switch(config-if-vlan12)#ip address 12.1.1.2 255.255.255.0

Switch(config)#interface vlan 14

Switch(config-if-vlan15)#ip address 14.1.1.1 255.255.255.0

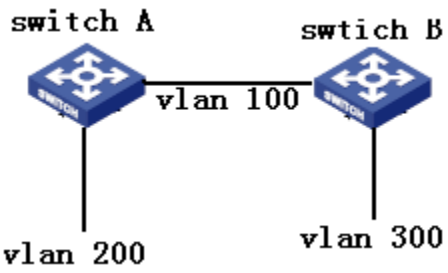
Switch(config)#ip route 15.1.1.0 255.255.255.0 12.1.1.1 bfd

When the link between Switch B and layer 2 switch is failing, Switch A can detect the change of Switch B immediately, here the static routing is at inactive state.

6.3.2 Example for Linkage of BFD and RIP Route

Example:

Switch A and Switch B are connected and run RIP protocol, both of them enable BFD function. When the link between Switch A and Switch B is failing, BFD can detect it immediately.



Configuration procedure:

Switch A:

Switch#config

Switch(config)#bfd mode active

Switch(config)#interface vlan 100

Switch(config-if-vlan100)#ip address 10.1.1.1 255.255.255.0

Switch(config)#interface vlan 200

Switch(config-if-vlan200)#ip address 20.1.1.1 255.255.255.0

Switch(config)#router rip

Switch (config-router)#network vlan 100

Switch (config-router)#network vlan 200

Switch(config)#interface vlan 100

Switch(config-if-vlan100) #rip bfd enable

Switch B:

Switch#config

Switch(config)#bfd mode passive

Switch(config)#interface vlan 100

Switch(config-if-vlan100)#ip address 10.1.1.2 255.255.255.0

Switch(config)#interface vlan 300

Switch(config-if-vlan300)#ip address 30.1.1.1 255.255.255.0

Switch(config)#router rip

Switch (config-router)#network vlan 100

Switch (config-router)#network vlan 300

Switch(config)#interface vlan 100

Switch(config-if-vlan100) #rip bfd enable

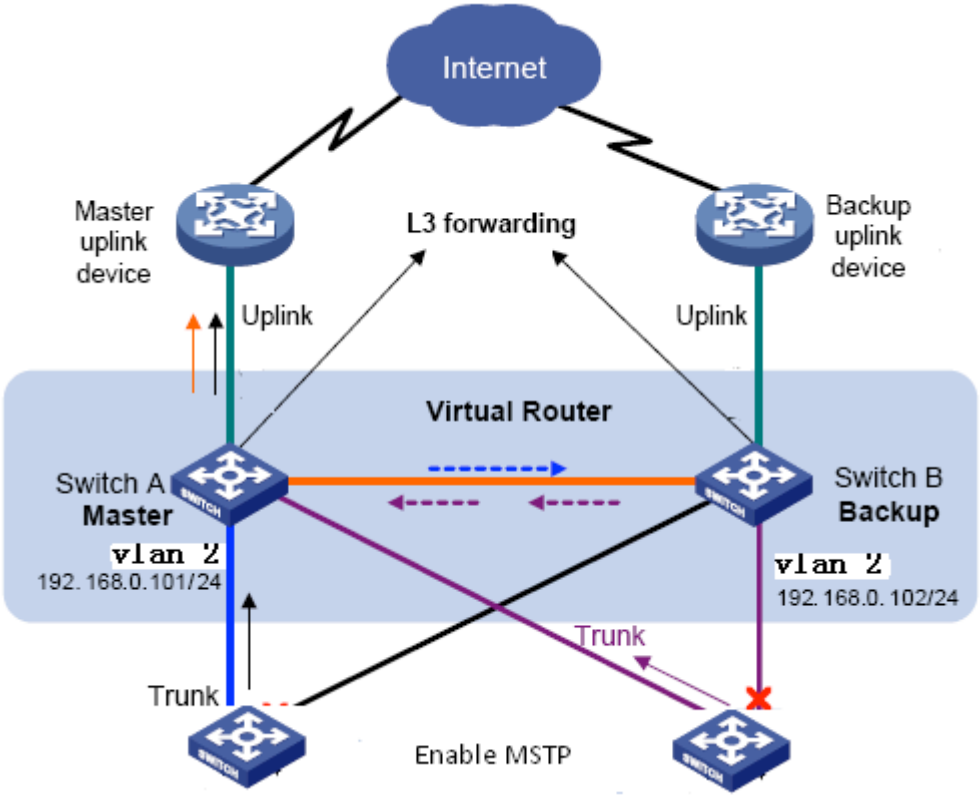
When the link between Switch A and Switch B is failing, BFD can detect it immediately and notifies RIP to delete the learnt route.

6.3.3 Example for Linkage of BFD and VRRP

Example:

When the master is failing, the backup cannot become the master until the configured timeout timer expires. The timeout is generally three to four seconds and therefore the

switchover is slow. To solve this problem, VRRP uses BFD to probe the state of the master. Once the master fails, the backup can become the new master within 100 ms.



Configuration procedure:

Configure Switch A

Switch#config

Switch(config)#bfd mode active

Switch(config)#interface vlan 2

Switch(config-ip-vlan2)#ip address 192.16.0.101 255.255.255.0

Switch(config)#router vrrp 1

Switch(config-router)#virtual-ip 192.168.0.10

Switch(config-router)#interface vlan 1

Switch(config-router)#enable

Switch(config-router)#bfd enable

Configure Switch B

Switch#config

Switch(config)#bfd mode passive

Switch(config)#interface vlan 2

Switch(config-ip-vlan2)#ip address 192.16.0.102 255.255.255.0

Switch(config)#router vrrp 1

```
Switch(config-router)#virtual-ip 192.168.0.10
Switch(config-router)#interface vlan 1
Switch(config-router)#enable
Switch(config-router)#bfd enable
```

6.4 BFD Troubleshooting

When the problem of BFD function happens, please check whether the problem is resulted by the following reasons:

- Check whether the route protocol neighbor is established successfully. If no route protocol neighbor is established successfully, here BFD can not process the detection.
- Check whether the configured source-ip is correct for linkage with static route, if the connectivity of IP between two peers fails, BFD can not process the detection.
- Check whether VRRP group is established successfully for linkage with VRRP protocol. If no VRRP group is established successfully, here BFD can not process the detection.