Content

Chapter 1 MPLS Overview

1.1 MPLS Overview

MPLS (Multiprotocol Label Switching), originating from IPv4, was first designed for improving the forwarding speed. Its core technology can be extended into multiple network protocols, including IPv6 (Internet Protocol version 6), IPX (Internet Packet Exchange), Appletalk, DECnet, CLNP (Connectionless Network Protocol) and etc, since the “Multiprotocol” in MPLS means supporting multiple protocols. MPLS technology is a combination of fast switch and L3 route forwarding hence can satisfy the network requirement of various new applications.

1.1.1 MPLS Introduction

Forwarding Equivalence Class

MPLS, as a class-based forwarding technology, will put packets with the same forwarding mode into a class named as FEC (Forwarding Equivalence Class). The same FEC group will be treated with the same way in MPLS networks. FEC is a group of L3 messages, which will be forwarded along the same path, at the same priority level, and in the same mode. There are two steps to finish the forwarding process:

- ☞ Analyze the packet header and divide packets into FEC
- ☞ Map the FEC to the next-hop

In traditional IP forwarding networks, each router will process the same packet with the above two steps. FEC can include one or more FEC units. All of them are L3 message packets that can be mapped to the same LSP.

At present, there are two types of FEC:

- ☞ Address Prefix: Use the Address Prefix to identify a FEC unit, whose length ranges from 0 to the full address length. Each Address Prefix FEC unit corresponds with a destination subnet.
- ☞ Host Address: Use the Host Address to identify a FEC unit, as each unit corresponds with a host address.

The division rules of FEC is very flexible, which can be any combination of source address, destination address, source port, destination port, protocol type, VPN and etc. For instance, in the traditional IP forwarding using the Longest Prefix Match Algorism, all packets targeted at the same destination address belong to one FEC.

Label

In MPLS networks,,each specific FEC will be encoded at the edge LSR into a label - a short, fixed-length value, which will be added to the head of packets and turn them into label packets, before they are forwarded. Besides a segment identifying FEC, labels also

include a COS segment, and thus representing FEC, precedence, and service class as a whole. LSR will divide packets reaching different ports into different FEC to establish the foundation of VPN. When a LSR creates a new FEC, it will also create a corresponding label, and advertise it to all peers. LSR maintain both incoming and outgoing labels. To implement load sharing, one FEC may correspond with multiple labels, but one label can only represent one FEC.

Labels, being carried in packet header, don't include topology information, and is only locally meaningful. The label length is 4 bytes. The figure demonstrates its encapsulation structure:

Fig 1-1 The Encapsulation Structure of a Label

There are 4 fields in a label:

- Label：The label value, whose length is 20 bits, a pointer for forwarding.
- Exp：3bits, used by QoS.
- S：1bit, the label's layered structure supported by MPLS, that is, there are multiple label layers. The value 1 represents the bottom-most layer of label.
- TTL：8bits, serves the same purpose as the TTL (Time To Live) in IP packets.

The label, like VPI/VCI of ATM and DLCI of Frame Relay, is identification for connections. If there is a label field in the link-layer protocol, such as VPI/VCI of ATM and DLCI of Frame Relay, the label will be encapsulated in these fields, otherwise, in a transitional layer between the link layer and the IP layer. Thus, labels can be supported by any link layer protocol.

Label Space

LSR can distribute a different label for a FEC according to its ingress port. As a result, packets from different ports can be forwarded independently, which is the basic foundation of VPN. To enhance the utilization efficiency of labels, MPLS provides the concept of label space, which is a label prefix. By allocating FECs belonging to different label spaces with the same label, the boundary of label is actually expanded. The label space is only meaningful when allocating labels, but not when forwarding them.

Label Switching

There is no need to analyze packet header in non-edge LSRs, instead, the label will be used as a pointer to the next-hop egress port and a new label. The label packet will replace the old label with the new one and then be forwarded through the specified egress port. Label switching will simplify and accelerate the forwarding process, and realize applications like VPN, QoS, traffic engineering and etc.

Label Switching Router

LSR (Label Switching Router) is the basic element of a MPLS network, with all LSRs supporting MPLS technology.

LSR is a device able to forwarding packets according to their label value. A LSR connecting an IP route network and a MPLS switching network is called an Edge LSR.

Such a LSR is able to adding labels to IP messages and forwarding data according to LSP, or deleting MPLS packet labels and forwarding data according to the IP routes. Each LSR must be distributed a global-alone LSR ID, usually get an interface IP address of LSR. Assume that, LSR Ru and Rd agree on the map between the label L and the FEC F. Packets can be forwarded from Ru to Rd based on the label L, in which case, Ru is the upstream LSR, and Rd the downstream LSR, that is to say, the forwarding of packets id always from the upstream LSR to the downstream one.

Label Switched Path

The path a FEC follows in the MPLS network is called a LSP (Label Switched Path). Two adjacent LSRs in a LSP are separately called the upstream and downstream LSR, along the direction of data transmission. In the next figure, R2 is the downstream LSR of R1,while R1 is the upstream LSR of R2.

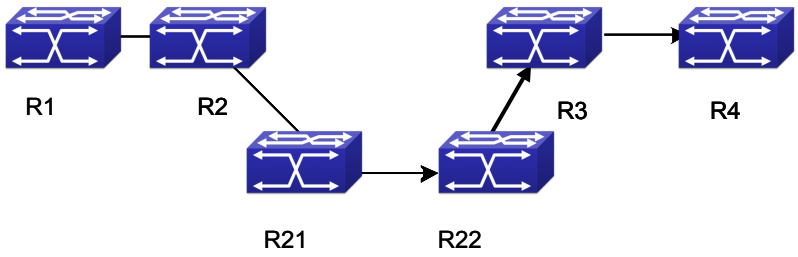

Fig 1-2 Label Switched Path LSP

The function of LSP, the same as the virtual circuit of ATM and Frame Relay, is a unidirectional path form the ingress of a MPLS network to its egress. Each router along the LSP is a LSR.

When downstream LSRs advertise labels to their upstream LSRs, all labels as a series and the LSR sequence compose a LSP. LSP will map the IP layer route information to a link layer switched path. LSP is a unidirectional packet forwarding path, along which, packets are always forwarded form an upstream LSR to a downstream one. To forward packets in the opposite direction, creating an entirely new and independent LSP is necessary. LSP always relates FEC with LSP. This relationship between FEC and LSP is called mapping packets to LSP.

1. The rules of mapping packets to LSP:
   a) If there is only one LSP, which includes a host-address FEC unit with the same destination address as the packet, map the packet to it;
   b) If there is more than one LSP satisfying condition 1, map the packet to any one of them.
   c) If there is only one LSP, whose address-prefix FEC unit can match the packet, map the packet to it.
   d) If there is more than one LSP satisfying condition 3, choose a LSP based on the Longest Prefix Match principle;
   e) If a packet will definitely pass through a specific egress LSR, and there is a LSP, the prefix FEC unit bounded to which is the address of that egress LSR, map

the packet to this LSP.

2. Additional Rules:
   a) If the destination address of the packet matches no LSP, the packet will be sent along the LSP with the same address as its Egress Router, as long as the LSP has an Address-prefix FEC unit.
   b) If a packet matches two LSPs, one of which includes a host-address FEC unit, and the other an address-prefix FEC unit, always map the packet to the first one.
   c) If the packet matches no LSP with a host-address FEC unit, it should not be sent along a LSP even if whose host-address FEC unit is the same as the packet's egress router address.
   d) The creation of LSP is based on connections, which are the result of topology information rather than the demand of data flow. That is to say, no matter data forwarded by this router exist or not, the LSP will always be created.

Label Merging

With the LSR mapping multiple incoming labels to the same FEC, all these incoming labels will correspond with the same outgoing label and egress port. As a result, when packets with different labels reach the LSR, all outgoing packets will carry the same label. This process is called Label Merging. Label Merging can decrease the label number in the MPSL domain, but maybe at the cost of losing ingress port information of the packets.

If the LSR doesn't support label merging, when there are multiple label requests, it will initiate a new label request to the downstream LSR once for each of them, no matter they have the same FEC or not. Otherwise, only one label request will be implemented.

Label Distribution Protocol

LDP (Label Distribution Protocol) is the MPLS control protocol, like singling protocols in traditional networks, whose function includes classifying FEC, distributing labels, creating and maintaining LSP and etc.

MPSL supports multiple label distribution protocols, including protocols specially designed for distributing labels, like LDP, CR-LDP (Constraint-Based Routing using LDP), and existing ones capable of it after extension, like BGP (Border Gateway Protocol), RSVP (Resource Reservation Protocol). Besides, manually configured static LSP is allowed.

LSP Tunnel Technology

MPLS supports LSP tunnel technology. Even if the path between an upstream LSR and a downstream LSR in a LSP is not provided by the routing protocol, MPLS allows creating a new LSP connecting the two, making them the start and end of it separately. This new LSP is a LSP tunnel, which avoids encapsulating the tunnel via traditional network layer.

If the routes passed by a tunnel are the same as those from the routing protocol, this tunnel is Hop-by-Hop Routed Tunnel; or, it is an Explicitly Routed Tunnel.

Multi-layer Label Stack

If a packet is transmitted in more than one layer of LSP tunnel, it will carry multiple layers of labels – Label Stack. At the ingress and egress of each tunnel, MPLS will PUSH

or POP a label accordingly.

The label stack follows the “Last-In-First-Out” principle, so MPLS will process labels from the stack top.

MPLS sets no limit to the label stack depth. If the label stack depth of a packet is m, the label at the stack bottom is level 1, and the one at the stack top will be level m. A packet without pushing any label will be treated as having an empty label stack (the label stack depth is 0).

1.1.2 MPLS Network Introduction

MPSL Network Structure

As demonstrated in the next figure, the basic unit composing the MPLS network is LSR; and a network consists of LSR is called a MPLS domain.

The LSR at the edge of a MPLS domain, connecting other customer networks is called LER (LER，Label Edge Router) , and the internal LSR is a core LSR. Core LSRs can either be routers supporting MPLS or ATM-LSR upgraded from ATM routers. LSRs in the domain communicate with each other via MPLS, while the MPLS domain edge is adapted via LER and traditional IP technologies.

Packets will be transmitted along a LSP composed of a series of LSRs after the ingress LER pushes a label to it. The ingress LER is called Ingress, egress LER called Egress, and routers in the middle called Trasit.

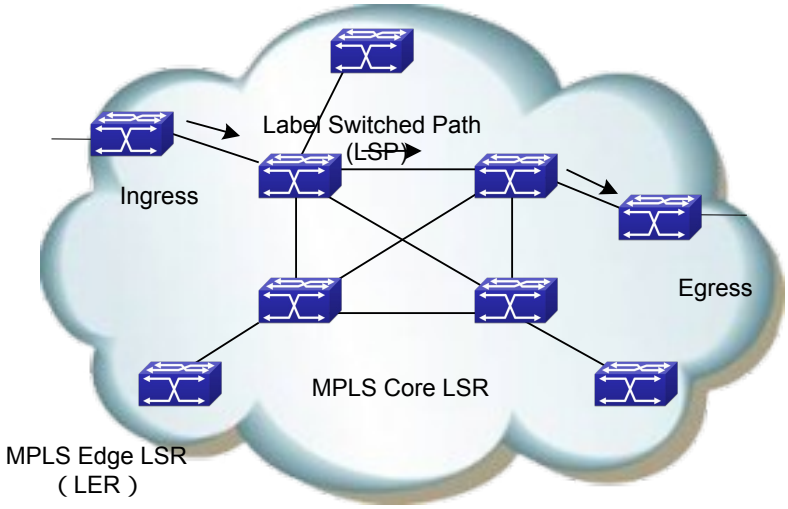

Fig 1-3 The MPLS Network Structure

The basic working process of MPLS based on the above figure :

First, LDP, together with traditional routing protocols (like OSPF, ISIS, etc) create route tables and LIB (Label Information Base) for FEC demanding services;

The ingress LER receives packets, completes L3 function, determines the FEC of the packets, labels them, and thus generates MPLS label packets.

Then, LSR in the network will forward packets according to their labels and LFIB (Label Forwarding Information Base) without implementing any L3 processing.

Finally, the egress LER of the MPLS will remove the label from the packet before the following IP forwarding.

To sum up, MPLS is neither a service or an application, but a tunnel technology, and a routing and switching technology platform integrated with label switching forwarding and network layer routing technology. This platform can support various high-level protocols and services with a certain guarantee of information security in the transmission.

1.1.3 Introduction to MPLS and Routing Protocols

When LDP creates LSP in hop-by-hop mode, it determines the next-hop based on the information from the forwarding table of each LSR along it. Since the information from forwarding tables are collected by routing protocols like IGP and BGP, LDP indirectly relates with them.

Besides, existing protocols like BGP and RSVP, can also distribute MPLS labels after extension.

Sometimes, it is necessary to extend some routing protocols in MPLS applications. For example, MPLS-based VPN requires extension to BGP, so that, BGP can distribute the VPN (Virtual Private Network) route information; MPLS-based TE (Traffic Engineering) requires extension to OSPF or IS-IS protocol, to carry link status information.

1.1.4 MPLS Application Introduction

MPLS technology originally combines L2 switching and L3 routing technology to enhance the route lookup speed. As ASIC (Application-Specific Integrated Circuit) develops, route lookup speed has no longer been the bottleneck of network development. As a result, MPLS's advantage in accelerating forwarding disappears.

However, combining the powerful L3 switching function of IP networks and efficient forwarding mechanism of traditional L2 networks, MPLS uses connection-oriented method at the forwarding plane, similar to the current L2 network. As a result, it can easily achieve seamless convergence of IP and L2 networks like ATM and Frame Relay, and provide better solutions for applications like QoS, TE and VPN.

MPLS-based VPN

Traditional VPN transmits private data in the public network via tunnel protocols like GRE, L2TP, and PPTP. Since LSP is a public network tunnel itself, MPLS is innately advantageous in implementing VPN.

MPLS-based VPN will connect different branches of a private network via LSP to form an integrated one. It also supports the intercommunication control between different VPN.

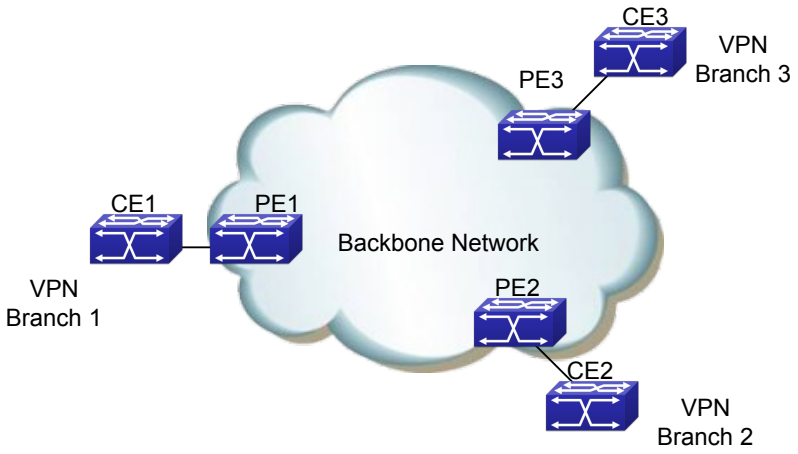

Fig 1-4 MPLS-based VPN

The above figure demonstrates the basic structure of MPLS-based VPN: CE (Customer Edge), a router, a switch or a host; PE (Provider Edge), in the backbone network.

PE manages VPN customers, establishes LSP connections between different PE and distributes routes to different branches of a VPN customer. The route distribution between PE is usually achieved via LDP or extended BGP.

MPLS-based VPN supports IP address multiplexing of different branches, and the intercommunication between different VPN. Different with traditional routes, VPN route contains extra identification of branches and VPN, making BGP extension a necessity, in order to carry VPN route information.

MPLS-based TE

MPLS-based TE and the Diff-serv feature can provide data flow at different precedent level with different service while ensuring a high network utility efficiency, and hence, be able to provide low-delay, low packet loss rate services with a guaranteed bandwidth to various data flows like voice and video.

Considering the difficulty of deploying TE over the whole network, the Diff-serv model is usually the method of implementing QoS in real networking resolutions.

The basic mechanism of Diff-Serv is mapping a service to a certain service class at the network edge, according to the required service quality. The service is uniquely identified via the DS segment (originated from ToS field) of IP packet. According to the segment, the routers in the backbone network will apply pre-configured service policy to different services, ensuring the service quality.

The service quality class mechanism and the label mechanism of Diff-Serv are similar to the label distribution mechanism of MPLS. In fact, the MPLS-based Diff-Serv is implemented via the combination of the DS distribution and MPLS label distribution.

1.1.5 MPLS PHP

In the MPLS network, the core LSR will forward packets according to their labels. The Egress router (Egress LER) will remove the label before implementing IP forwarding.

In fact, in simple MPLS applications, where the Egress routers only implement IP forwarding, labels will become useless. In such cases, popping the labels out via the Penultimate Hop Popping feature at the penultimate router will stop the Egress router from processing the labels.

Chapter 2 LDP

2.1 LDP Introduction

LDP protocol is used for label distribution in the MPLS label switching environment, and only applies to networks capable of label switching. LDP, integrated with traditional routing algorithm, distribute labels, advertise <label, FEC> map, create and maintain Label Forwarding Information Base and LSP, by transmitting various messages via TCP connections. LDP is used to distributing public network label in the MPLS VPN environment.

LDP doesn't create any route; instead, it obtains routes from the system, distributes labels for them and advertises the labels to its upstream router. At the same time, for the FEC having a downstream, LDP will receive a label from the downstream, take it as the outgoing label and create a label switched path, which means to create an entry of switching the incoming label as the outgoing one. If the label distributed by the downstream is 3, the LDP will create an entry of popping out the label.

LDP is defined in RFC3036; and its latest standard is RFC5036. It switches the map between labels and routes via the TCP connection between peers. Two neighbor discovery modes are supported by LDP: the basic mode (automatic discovery) and the extended mode (specified). The automatic discovery of peers is implemented via the UDP multicast messages to all routers (224.0.0.2), using the port 646 in both TCP and UDP messages.

The main process is as follows:

- ☞ Discover and maintain neighbors: after LDP is global enabled and interface enabled, it will send multicast Hello messages on the specified interface (unless it disables the multicast-based neighbor discovery) to advertise the network about its existence. The Hello messages will carry its transmission address, the address for TCP connections. The adjacency will be created when receiving Hello messages from other LSRs, and maintained by periodically sending Hello messages.
- ☞ Establish and maintain sessions: LDP sessions are TCP-based; First, compare the transmission address in the Hello message from the other end and that from this end, set the one with bigger value as ACTIVE and the other PASSIVE. The ACTIVE router will initiate a connect request to establish a TCP connection (to avoid the similar connection conflict problems suffered by BGP neighbors). Once the TCP connection is established, the two parts will send initialization messages to negotiate session parameters. A session will be established once the negotiation succeeds. After that, the two neighbors will send the local interface address list and label information to each other. To hold the connection when there is no data, KEEPALIVE messages will be sent.

- ☞ Create and maintain LSP: a session is necessary for each pair of LSR peers to switch label information, which create LSP by switching FEC and label binding messages.
- ☞ Cancel sessions: Without any message from the other end for a long time, LDP will disconnect the session and notify the close of the session to the other end by sending a notification messages.

Please notice that, LDP won't distribute labels for default routes, or BGP routes (unless explicitly specified).

2.1.1 Basic Concept of LDP

LDP Peer

When distributing labels to FEC, LDP needs to advertise this label and its meaning in the MPLS network to create LSP. LSR is a LDP peer when switching label information via LDP. LDP peers obtain each other's label map and other messages.

LDP Session

Two LSR will create a LDP session between each other after exchanging LDP Discovery Hello messages. LSP relies on LDP sessions to exchange messages like label map, release.

Two steps to establish a LDP session

- ☞ Establish the transmission connection.
- ☞ Initialize the session

Two types of LDP session:

- ☞ Local LDP Session: the two LSR establishing the session are directly connected.
- ☞ Remote LDP Session: the two LSRs establishing the session are indirectly connected

LDP Message Type

Four types of LDP messages:

- ☞ Discover message: to advertise and maintain the existence of LSR in the network;
- ☞ Session message: to create, maintain and terminate the sessions between LDP peers;
- ☞ Advertisement message: to create, change and delete the map from label to FEC;
- ☞ Notification message: to provide advice messages and error notices.

To ensure the reliable sending of LDP messages, LDP uses TCP to send Session, Advertisement and Notification messages, and UDP to send Discovery ones.

2.1.2 Introduction to LDP Message Format

LDP PDU

LDP PDU includes a LDP header and several LDP messages. The LDP header

format is as follows:

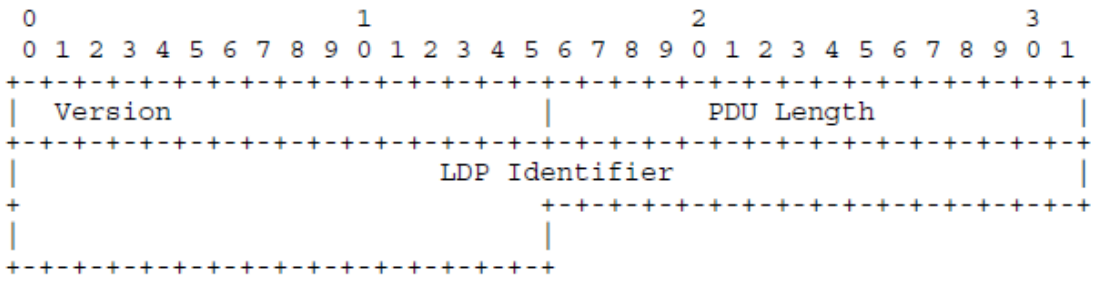

Fig 2-1 The LDP Header Format

- Version：The LDP version, I byte. The current LDP version is 1.
- PDU Length：The total length of the LDP message (in byte), 2 bytes.
- LDP ID：LDP ID, 6 bytes. The first 4 bytes is the globally unique LSR ID, and the rest 2 are label space ID, which is 0 when it comes to the global label space.

TLV Encoding

LDP encapsulates parameters in LDP messages via TLV (Type-Length-Value). The LDP TLV format is as follows:

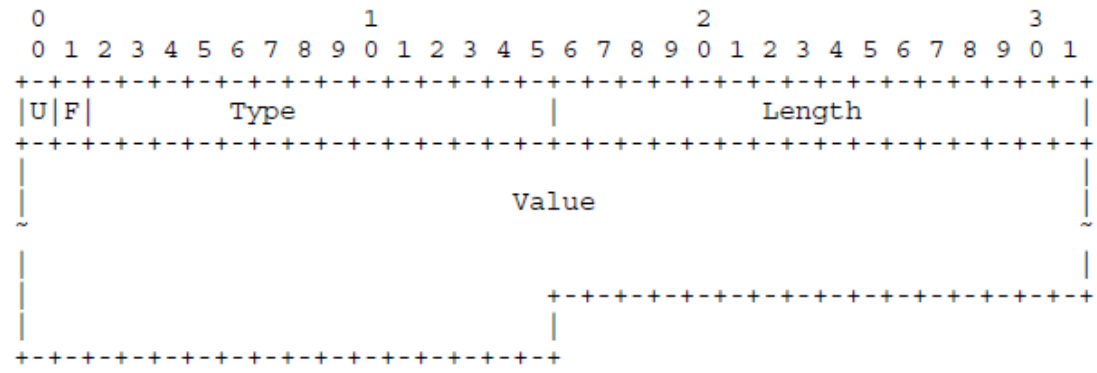

Fig 2-2 The TVL Format of LDP

- U bit：Unknown flag, I bit. If the U flag is 0, LSR should notify the source LSR of the packet and ignore the whole message; otherwise, ignore this TLV parameter and analyze other ones normally.
- F bit：Forwarding unknown TLV flag, 1bit. This flag only applies to LDP messages with unknown TLV and a U bit set as 1. If the F flag is 0, stop forwarding unknown TLV parameters; otherwise, forward them;
- Type：Type, 14 bits.
- Length：Length, 1 byte. The length of TLV value segment.
- Value：The Value segment, whose length is defined by the parameter of “Length”.
- The Value segment of TLV can also contain TLV parameters, meaning that, TLV are embeddable. The first byte of TLV doesn’t need alignment.

Currently defined TLV types:

TLV	Type
FEC	0x0100
Address List	0x0101
Hop Count	0x0103
Path Vector	0x0104
Generic Label	0x0200
ATM Label	0x0201
Frame Relay Label	0x0202
Status	0x0300
Extended Status	0x0301
Returned PDU	0x0302
Returned Message	0x0303
Common Hello Parameters	0x0400
IPv4 Transport Address	0x0401
Configuration Sequence Number	0x0402
IPv6 Transport Address	0x0403
Common Session Parameters	0x0500
ATM Session Parameters	0x0501
Frame Relay Session Parameters	0x0502
Label Request Message ID	0x0600
Vendor-Private	0x3E00- 0x3EFF
Experimental	0x3F00- 0x3FFF

2.1.3 LDP Label Management

In the MPLS system, the downstream LSR determines the distribution of label to specific FEC, and notifies the upstream. That is to say the labels are specified by the downstream and distributed from downstream to upstream.

Label Advertisement Mode

In the MPLS domain, packets will be forwarded to the downstream LSR with the downstream LSR label after the label switching process in the upstream LSR. The FEC labels distributed by the downstream LSR apply only to itself and the upstream LSR, and should be advertised to the upstream LSR. MPLS defines two label advertisement modes for the downstream LSR passing labels to its upstream LSR:

- ☞ DoD (Downstream On Demand) : LSR only distributes and advertises a label for the specified FEC after receiving a label request message from the upstream.
- ☞ DU (Downstream Unsolicited) : LSR distributes and advertises a label for the specified FEC without receiving a label request message from the upstream. It will automatically send label map information and notify the upstream LSR.

These two modes can be mixed, with each LSR interface configured independently to use one of them. During initialization, the upstream and downstream LSR have to

exchange their label advertisement mode information to reach an agreement on the mode. Otherwise the creation of LSP will fail.

The figure demonstrates the process of LDP label advertisement:

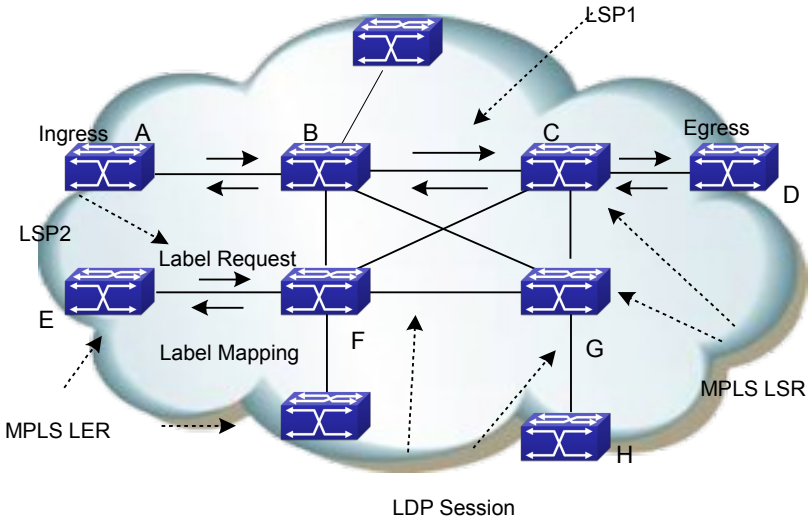

Fig 2-3The Process of Label Advertisement

For example, as for LSP1 in the above figure, LSR B is the upstream LSR of LSR C, while LSR C is the downstream LSR of LSR B.

The main difference of two label advertisement mode lies on whether the label advertisement is DoD or DU.

The following is the detailed label advertisement process of these two modes:

(1) DoD (downstream-on-demand)

The upstream LSR send a Label Request Message, which carries FEC description to its downstream LSR. The downstream LSR will distribute a label for this FEC, and respond to the upstream with the mapped label via a Label Mapping Message.

When will the downstream LSR respond the Label Mapping Message depends on the label advertisement mode it adopted.

1) In Ordered mode, it will send the Label Mapping Message to the upstream only after receiving a Label Mapping Message from its downstream.
2) In Independent mode, it will immediately send the Label Mapping Message to the upstream no matter it receives a Label Mapping Message from its downstream or not.

Usually, the upstream LSR chooses the downstream LSR based on the routing table. In the above figure, all LSR along the LSP1 work in the ordered mode, while LSR F in LSP2 in the Independent mode.

(2) DU (downstream unsolicited)

The downstream LSR will automatically advertise the label mapping message to its upstream LSR after the LDP session successfully created. The upstream LSR will save

the message and process it according to its retention mode.

Label Distribution Control Mode

In the MPLS domain, LSR generate a LSP from the ingress router and the egress router via switching labels, based on the route-forwarding path created by IGP in the MPLS domain. Only a complete path is useful for pack forwarding. The creation of LSP is the LSR label advertisement process; hence, controlling the creation of LSP is controlling the LSR label advertisement. MPLS defines two LSP control modes to determine when the downstream LSR will advertise labels to the upstream LSR:

- ☞ Ordered Mode: For a FEC label mapping of a LSR, the LSR only advertise the mapping to its upstream when it already has the label mapping of the FEC next-hop, or when it is the egress router of the FEC. The label advertisement of a flow starts from the egress router of this FEC flow, binding routers from downstream to upstream, thus to guarantee the mapping between labels and the flow is complete and coherent in the whole network. The ordered mode can prevent loop more effectively.
- ☞ Independent Mode: LSR doesn't have to wait for the label of the FEC next-hop to advertise labels to its peer. It can notify label mapping to the LSR connected to it at any time. This mode may cause the LSR advertise a label to its upstream before receiving one from its downstream. This mode can accelerate the creation and aggregation of LSP.

Requirements for LSR to be an Egress router:

- ☞ The FEC quotes the LSR address;
- ☞ The FEC next-hop router locates outside the label switching network;
- ☞ The FEC unit passes the route area, such as another OSPF SUMMAERY domain, or another autonomy system of OSPF, BGP.

Label Retention Mode

Label Retention Mode determines how the LSR handles the currently useless mapping from label to FEC it received. In DU mode, the upstream LSR may receive a large number of <FEC, label> map sets from the downstream LSR, in which case, only when the FEC in the map set is the local FEC next-hop of the upstream LSR, this map set is meaningful for the label forwarding. MPLS defines two label retention modes to determine the processing of currently useless map set.

- ☞ Conservative Mode: the LSR will reserve the label mapping received from the neighbor LSR no matter the neighbor is its next-hop or not. The advantage of this mode is that it only creates and maintain the labels that meaningful for data forwarding, a very significant feature when the label space is limited (ATM switching).
- ☞ Liberal Mode: the LSR only save label maps from the neighbor LSR which is its next-hop. The advantage of this mode is that the expense of processing route changes is very low; and the disadvantage is many useless labels will be advertised and maintained.

In the Liberal label retention mode, LSR can adapt rapidly to route changes; in the Conservative mode, LSR can distribute and save relatively less labels. The Conservative retention mode, together with the DoD mode, usually applies to LSR with limited label

space.

Some Basic Concepts of Label Switching

- ☞ NHLFE: Next Hop Label Forwarding Entry. It is used to describe the operation to the label, including Push and Swap.
- ☞ FTN (FEC to NHLFE map): the process of mapping FEC to NHLFE on the Ingress router.
- ☞ ILM (Incoming Label Map): the process of mapping received labels to NHLFE by LSR.

The Label Switching Process

The Ingress LER divides the packets entering the network into FECs. The packets belonging to the same FEC will follow the same path - LSP, in the MPLS domain. LSR will distribute a label for the incoming FEC packet and forward it through the corresponding interface.

The detailed process of label switch is as follows:

- ☞ All LSRs along the LSP will create an ILM first, the entries in which are the rule of mapping the incoming labels.
- ☞ LSR will map the labels of received packets to NHLFE;
- ☞ LSR will find the corresponding NHLFE in the LIB based on the label, replace it with the new label and then forward the label packet.

2.1.4 LDP Session

There are four steps to establish a LDP session:

- ☞ Discover
- ☞ Establish and maintain the session
- ☞ Create LSP
- ☞ Cancel the session

Discover

At this step, the LSR will send Hello messages periodically to adjacent LSRs, notifying them about its existence, in order to establish a session. In the basic discover mechanism, LSR will discover its LDP peers automatically via this process without manual configuration. There are two discover mechanisms:

- ☞ Basic Discover Mechanism

The Basic Discover Mechanism is used to discover local LDP peers – LSRs directly connected via the link layer, and create a local LDP session. In this mode, the LSR will send LDP Link Hello messages periodically via UDP messages to the multicast address marked as “all routers in the subnet”.

LDP Link Hello messages carry the LDP ID of the interface and other related information. If the LSR receives a LDP Hello Message at an interface, it means that there is a LDP peer at this interface (Link Layer).

- ☞ Extended Discover Mechanism

The extended discover mechanism is used to discover remote LDP peers – LSRs not directly connected via the link layer, and created remote LDP sessions. In this mode,

the LSR will send LDP Targeted Hello messages periodically to the specified IP address via UDP messages.

LDP Targeted Hello messages carry the LDP ID of the interface and other related information. If the LSR receives a LDP Targeted Message at an interface, it means that there is a LDP peer at Network Layer.

Establish and Maintain the Session

After discovering a LDP peer, LSR will began to establish the session in two steps:

- ☞ Establish the transmission layer connection, that is, a TCP connection between LSRs;
- ☞ Initiate the session between the LSRs, negotiate all concerning parameters, such as the LDP version, the label advertisement mode, the timer value, the label space. After the negotiation succeeds, the session is established between the LSRs.

The session will be maintained by Keepalive messages after established.

Create LSP

The process of creating LSP is mapping FEC and labels and advertising the maps to the adjacent LSRs along the LSP, which is realized via LDP. Take DoD mode as the example, the main steps are as follows:

(1) When the network routes change, if an edge router finds out a new destination address in its route table which belongs to none of the existing FECs, it needs to create a new FEC for this destination address. The edge LSR determines the route for the FEC, initiates a label request message to its downstream LSR, and specifies for which FEC this label request is.

(2) The downstream receiving the label request message will save this message, finds the corresponding FEC next-hop according to the local route table and then sends a label request message to its downstream.

(3) When the label request message reaches the destination router or the egress router of the MPLS network, if the router has available labels, and judges the label request messages as legal, it will distribute a label for the FEC, and send a label mapping message containing the label information to its upstream;

(4) The LSR receiving the label mapping message will check the state of label request messages saved locally. If there is a corresponding label request message of a FEC label mapping message in the data base, LSR will distribute a label for the FEC, and add a new entry in its LFIB, and then send the label mapping information to its upstream.

(5) When the ingress LSR receives a label mapping message, it also should add a corresponding entry in its LFIB, and thus finish the creation of LSP.

Cancel the session

LDP maintains adjacency by checking Hello messages. It also maintains session by checking Keepalive messages. If there is no Keepalive message received within a certain period of time, the LDP session will close the connection.

Each LDP session can include one or more Hello adjacencies. LDP maintains Hello adjacency via periodical Hello messages. If there is no LDP Discovery Hello message received within a certain period of time, the LDP session will close the Hello connection.

When closing the last Hello adjacency in the LDP session, LDP will send notification messages, and close the transmission connection.

2.1.5 LDP Loop Detection

Creating LSP in the MPLS domain also needs to prevent loops. The LDP loop detection mechanism can detect LSP loops and avoid them.

To detect loops in the MPLS domain, all LSRs should be enabling the loop detection. But when establishing LDP sessions, the configurations of loop detection on the two parties don't have to be the same.

There are two LDP loop detection modes:

The maximum hop count

It is the number of LSR passed by the label messages (including label mapping and label request). When LSR transmits label information with the hop-count parameter, it will first increase the hop count by 1. When the hop count reaches the configured maximum value, it means that a loop exists, and the LSP creation will fail. If the hop count is 0, it means the hop count is unknown. The hop count of label messages is always 0. The default maximum hop count is 255.

Path Vector

It is used to record the path information in label mapping or label request messages. At each hop, the LSP checks whether its LSR ID is in the record. The following two conditions mean the existence of a loop and the failure of the LSP creation.

- ☞ There is a record of this LSR in the path vector record;
- ☞ The hop count of the path exceeds the configured maximum value.

If no record of its LSR ID is found, a new one will be added. The maximum value of path vector is the same as that of the hop count.

2.2 LDP Configuration

LDP Configuration Task Sequence:

1. Enable MPLS Globally (Necessary)
2. Enable LDP (Necessary)
   (1) Enable/Disable the LDP module
   (2) Enable/Disable label-switching on the interface
   (3) Enable/Disable LDP module on the interface
3. Configure the LDP parameters (optional)
   (1) Configure the LDP label management mode
      1) Configure the LDP label retention mode
      2) Configure the LDP label advertisement mode
      3) Configure the LDP label control mode
   (2) Configure the LDP loop detection
      1) Enable/Disable the LDP loop detection

2) Set the maximum hop count of the LDP loop detection

(3) Configure the LDP specified peers

(4) Configure other LDP parameters

1) Configure the aging time or interval of each timer

2) ID Configure the LDP router ID

3) Configure the TCP interface address of LDP

4) Configure the LDP to discover peers via multicast Hellos or not.

5) Configure the LDP to import BGP routes or not.

6) Enable/Disable the LDP label merging capability

7) Configure the LDP to transmit release messages or not.

8) Configure the LDP to retry or not when the label request is rejected.

9) Hello Configure the LDP to receive Hello from specified targets

4. Clear LDP connections or adjacencies.

1. Globally enable MPLS

Command	Explanation
Global Mode	
mpls enable **no mpls enable**	necessary Enable MPLS; the no operation will disable MPLS.

2. Enable LDP

It is easy to implement basic configurations of LDP in Digitalchina. Usually users only have to enable the LDP switch, and enable it on the interface where the LDP will work. Please notice that, the interface with LDP enabled should enable label switching.

Command	Explanation
Global Mode	
router ldp **no router ldp**	Necessary LDP Enable/disable LDP; disabled by default
Interface Configuration Mode	
label-switching **no label-switching**	Necessary Enable/disable label-switching; disabled by default
mpls proxy loopback-group <1-max_agg_num> **no mpls proxy loopback-group**	Enable MPLS proxy, the default does not enable the function, enable MPLS proxy when the boardcard of MPLS ingress unicast packet does not support MPLS.
ldp {enable|disable}	Necessary LDP Enable/disable LDP on the interface; disabled by default

3. Configure the LDP parameters

(1) Configure the LDP label management mode

1) Configure the LDP label retention mode

2) Configure the LDP label advertisement mode

3) Configure the LDP label control mode

Command	Explanation
Router Configuration Mode	
label-retention-mode {conservative| liberal}	Optional Configure the global label retention mode: Conservative or Liberal; it is liberal by default
advertisement-mode {downstream-on-demand|downstream-unsolicited}	Optional Configure the global label advertisement mode: **downstream-on-demand** or **downstream-Unsolicited.** This mode relates with the other two. The change of it will change the label retention mode and the global label path control mode at the same time. It is **downstream-unsolicited** by default
control-mode {ordered|independent}	Optional Configure the global label retention mode: Ordered or **independent**; it is **independent** by default
Interface Configuration Mode	
ldp label-retention-mode {conservative | liberal}	Optional Configure the label retention mode of the interface; the default value is the same as the global configuration. If the configuration differs with the global one, the interface configuration will take effect.
ldp advertisement-mode {downstream-on-demand|downstream-unsolicited}	Optional Configure the label advertisement mode of the interface; the default value is the same as the global configuration. If the configuration differs with the global one, the interface configuration will take effect.

(2) Configure LDP loop detection

1) Enable/disable LDP loop detection

2) Configure the maximum hop count of LDP loop detection

Command	Explanation
Router Configuration Mode	
[no] loop-detection	Optional Enable LDP loop detection, the no operation will disable it.
[no] loop-detection-count <*count*>	optional Configure the maximum hop count of LDP loop detection, whose default value is 255, the no operation will restore the default value.

(3) Configure the LDP specified peers

Command	Explanation
Router Configuration Mode	
[no] targeted-peer <*ip-addr*>	optional Configure the remote peer of the LDP targeted destination.

(4) Configure other LDP parameters

1) Configure the aging time or interval of each LDP timer
2) ID Configure LDP router ID
3) Configure the TCP interface address of LDP
4) Configure the LDP to discover peers via multicast Hellos or not,
5) Configure the LDP to import BGP routes or not.
6) Configure the LDP to enable label merging capability or not.
7) Configure the LDP to transmit release messages or not.
8) Configure the LDP to retry or not when the label request is rejected
9) Hello Configure the LDP to receive Hello from the specified targets

Command	Explanation
Route Configuration Mode	
[no] keepalive-interval <*interval*>	Optional Configure the interval of sending LDP keepalive messages, whose default value is 10 seconds; the no operation will restore the default value
[no] keepalive-timeout <*time-val*>	Optional Configure the LDP keepalive timeout, whose default value is 30 seconds; the no operation will restore the default value
[no] Hello-interval <*Hello-interval*>	Optional Configure the interval of sending multicast HELLO messages, whose default value is 5 seconds; the no operation will restore the default value
[no] hold-time <*hold-time* >	Optional Configure the LDP multicast peer hold time, whose default value is 15 seconds; the no operation will restore the default value
[no] targeted-peer-Hello-interval <*Hello -interval*>	optional Configure the interval of sending HELLO to specified targets, whose default value is 15 seconds; the no operation will restore the default value

[no] targeted-peer-hold-time <*hold-time*>	optional Configure the LDP targeted peer hold time, whose default value is 45 seconds; the no operation will restore the default value
Interface Configuration Mode	
[no] ldp keepalive-interval <*interval*>	optional Configure the interval of sending LDP keepalive messages on a specified interface; the no operation will restore the default value
[no] ldp keepalive-timeout <*time-val*>	Optional Configure the LDP keepalive timeout on a specified interface; the no operation will restore the default value
[no] ldp Hello-interval <*Hello-interval*>	Optional Configure the interval of sending LDP multicast HELLO messages on a specified interface; the no operation will restore the default value
[no] ldp hold-time <*hold-time*>	optional Configure the LDP multicast peer hold time on a specified interface; the no operation will restore the default value
[no] ldp targeted-peer-Hello-interval <*Hello-interval*>	optional Configure the interval of sending LDP HELLO messages to specified targets on a specified interface; the no operation will restore the default value
[no] ldp targeted-peer-hold-time <*hold-time*>	optional Configure the LDP targeted peer hold time on a specified interface; the no operation will restore the default value
router configuration mode	
[no] router-id <*ip-addr*>	optional Configure the LDP router ID, which is obtained automatically by default. The no operation will cancel the manually configured router ID, and automatically obtain a valid interface IP address as the router ID.

[no] transport-address <*ip-addr*>	optional Configure the IP address of LDP for TCP connections. Please notice that this address has to be that of a loopback interface on the main VRF. The no operation will cancel the manual configuration and let LDP automatically choose the TCP address
[no] multicast-Hellos	optional Configure the LDP to discover peers via multicast HELLOs, the no operation will do the opposite. Using multicast HELLO is the default setting.
[no] import-bgp-routes	Optional Configure the LDP to import BGP routes; the no operation will do the opposite. Not importing BGP routes is the default setting.
[no] global-merge-capability {merge-capable|non-merge-capable}	optional Configure the LDP to enable global label merging capability or not, the no operation will restore the default value.
[no] propagate-release	optional Configure the LDP to advertise label release messages to peers, the no operation will do the opposite. Not transmitting label release messages is the default setting.
[no] request-retry	optional Configure the LDP to retry 5 times when the label request is rejected, the no operation will disable the retry.
[no] request-retry-timeout <*time-val*>	optional Configure the retry interval, whose default value is 5 second, the no operation will restore the default value.
[no] targeted-peer-Hello-receipt	optional Configure LDP to receive HELLOs from specified targets, even the targeted peer is not configured on the host. Not receiving such HELLOs is the default setting. The no operation will restore the default configuration. Please notice that, if targeted LDP peers are configured, targeted-peer-Hello-receipt should be too.

4. Clear the LDP connections or adjacencies

Command	Explanation
Admin Mode	
clear ldp adjacency {<*ip-addr*>|*}	Optional Clear specified LDP adjacencies, "*"means all.
clear ldp session {<*ip-addr*>|*}	optional Clear specified LDP sessions, "*"means all.

2.3 LDP Typical Instances

Some designations of LDP are for adapting different network environments. Its configuration is very simple in the typical Ethernet environment. Due to the development of hardware system, especially the popularity of L3 switches, the pure MPLS network has already lost its importance to MPLS VPN.

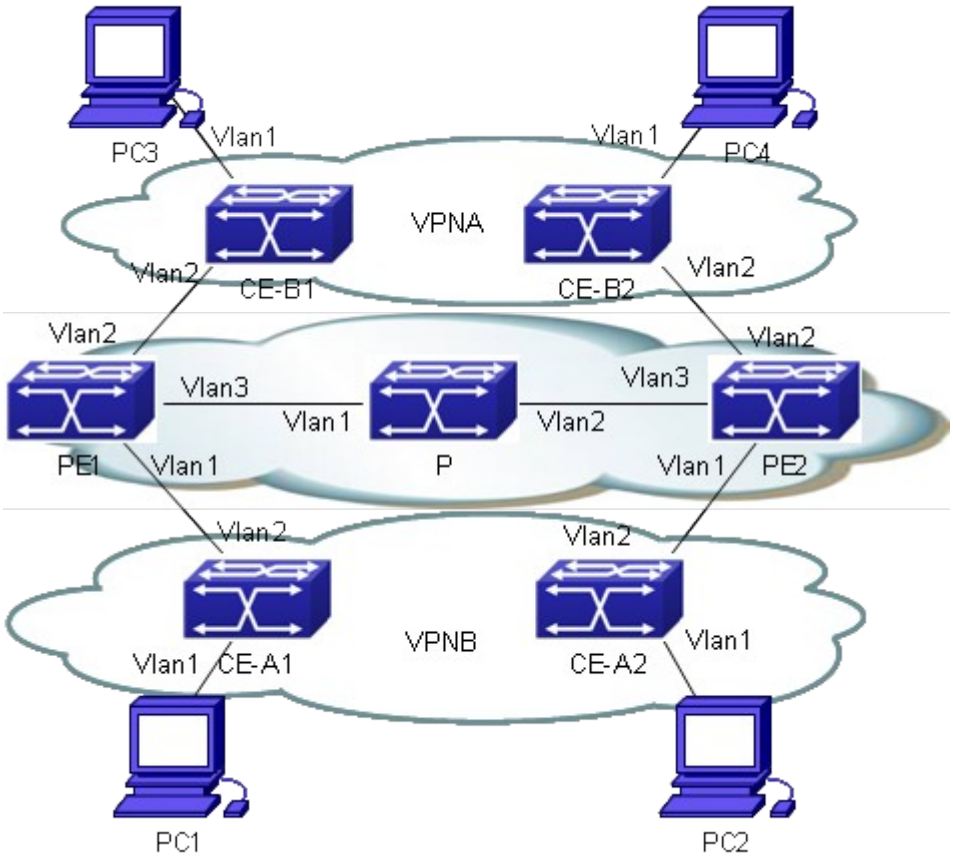

Fig 2-4 MPLS VPN Typical Instance

The above figure demonstrates a typical MPLS VPN instance, in which, PE1, P and PE2 form the public network area – the area switching via MPLS. CE-A1 and CE-A2 form VPN-A, CE-B1 and CE-B2 form VPN-B. Both VPNs communicate via the public network

label switching, and need to configure LDP for distributing and advertising labels in the public network area. To guarantee the reachability of routes, we advertise routes via OSPF.

The LDP configuration of PE1 is as follows:

```
PE1#config
PE1(config)#mpls enable
PE1(config)# router ldp
PE1(config-router)#exit
PE1(config)#interface vlan 3
PE1(config-if-Vlan3)#ip address 202.200.1.2 255.255.255.0
PE1(config-if-Vlan3)#ldp enable
PE1(config-if-Vlan3)#label-switching
PE1(config-if-Vlan3)#exit
PE1(config)#router ospf
PE1(config-router)#network 200.200.1.1/32 area 0
PE1(config-router)#network 202.200.1.0/24 area 0
PE1(config-router)#exit
```

The LDP configuration of P is as follows:

```
P#config
P(config)#mpls enable
P(config)# router ldp
P(config-router)#exit
P(config)#interface vlan 1
P(config-if-Vlan1)#ip address 202.200.1.1 255.255.255.0
P(config-if-Vlan1)#ldp enable
P(config-if-Vlan1)#label-switching
P(config-if-Vlan1)#exit
P(config)#interface vlan 2
P(config-if-Vlan2)#ip address 202.200.2.1 255.255.255.0
P(config-if-Vlan2)#ldp enable
P(config-if-Vlan2)#label-switching
P(config-if-Vlan2)#exit
P(config)#router ospf
P(config-router)#network 202.200.1.0/24 area 0
P(config-router)#network 202.200.2.0/24 area 0
P(config-router)#exit
```

The LDP configuration of PE2 is as follows:

```
PE2#config
PE2(config)#mpls enable
PE2(config)# router ldp
PE2(config-router)#exit
PE2(config)#interface vlan 3
PE1(config-if-Vlan3)#ip address 202.200.2.2 255.255.255.0
PE2(config-if-Vlan3)#ldp enable
```

PE2(config-if-Vlan3)#label-switching
PE2(config-if-Vlan3)#exit
PE2(config)#router ospf
PE2(config-router)#network 200.200.1.2/32 area 0
PE2(config-router)#network 202.200.2.0/24 area 0
PE2(config-router)#exit
Please refer to BGP VPN typical instances for the configuration of BGP.

2.4 LDP Troubleshooting

When configuring and using LDP, some problems like incorrect physical connections, configuration errors may cause it to fail, so plesase pay attention to the following notices to avoid them:

- ☞ First, make sure the system enables LDP globally and on the active interface. Notice that the LDP can only be enabled on interfaces after it is enabled globally.
- ☞ Second, use the “show ldp interface” command to check whether the LDP has been enabled correctly on the interface after the connection succeeds. If the LDP has been correctly enabled but cannot be displayed, it is possible that the interface is not in the UP mode or not configured with interface label-switching.
- ☞ Then, make sure the adjacent interfaces are in the same segment, and check whether the LDP can discover peers and establish adjacencies with them normally via the “show ldp adjacency” command. If no peer is discovered or no adjacency is established, it is possible that the interfaces may belong to different segments, or one of the local host and its remote neighbor disables multicast HELLO. Besides, when establishing TCP connection, LSR ID is the default address, as, please make sure advertise the LSR ID route to the remote end.
- ☞ Check whether the state of LDP session with “show ldp session” is operational, since only in this state, LDP sessions can switch messages. If the LDP session can’t be established, use “show ldp” to check the TCP addresses of the two parties, and lookup the route table to make sure the route of the remote end is reachable.

At last, given all above steps succeed, use “show ldp fec” to check the routes imported by LDP and their information, or check the created entries with “show mpls ftn” and “show mpls ilm”.

Chapter 3 MPLS VPN

3.1 BGP/MPLS VPN Introduction

3.1.1 BGP/MPLS VPN Network Structure

BGP/MPLS VPN is a PE-based L3VPN technology in the VPN solutions provide by providers, using BGP to advertise VPN routes and MPLS to forward VPN messages in the provider backbone network.

The BGP/MPLS VPN networking is flexible, extendible, and can support MPLS QoS and MPLS TE conveniently, resulting in its increasingly popular application.

BGP/MPLS VPN model consists of three parts: CE, PE and P.

- ☞ P router: Provide Router. It locates in the MPLS domain, and is able to switch fast-forwarding MPLS data flow based on labels. P router receives MPLS messages, switch labels and then output them.
- ☞ PE router: Provide Edge Router. It locates at the edge of the MPLS domain, for converting IP messages and MPLS messages. PE router receives IP messages, pushes MPLSU labels, and output MPLS messages; or receives MPLS messages, pop labels, and output IP messages. On PE routers, the ports connected with other P routers or PE routers are “public network port”, configured with public network IP address; those connected with CE routers are “private network port”, configured with private network address.
- ☞ CE router: Customer Edge Router. It locates at the edge of the customer IP domain, connected directly to PE route, for aggregating customer data and forwarding route information of the customer IP domain to PE router.

The next figure demonstrates a BGP/MPLS VPN networking:

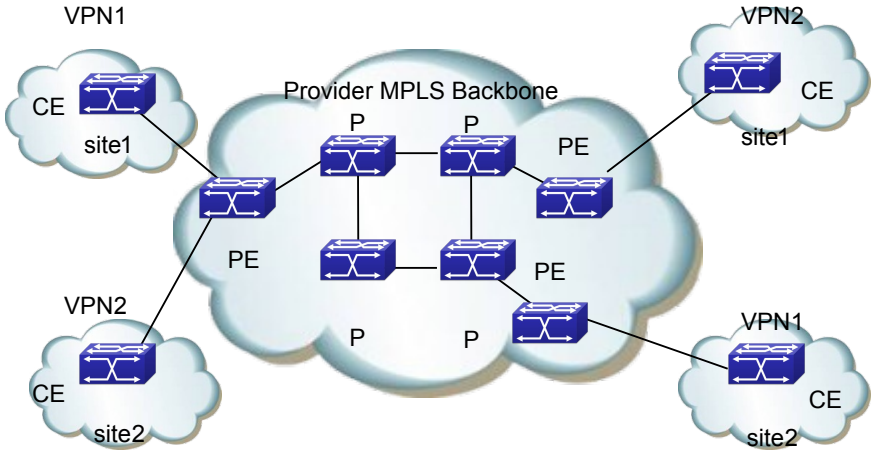

Fig 2-1 BGP/MPLS VPN Networking

The division of CE and PE is based on the management areas of SP and customers, since CE and PE are the edge between the two areas.

CE is usually a router. When the adjacency between it and the PE directly connected to it, CE will advertise the local VPN route to PE, and learn the remote VPN route from PE. CE and PE use BGP/IGP to exchange route information or static routes.

PE will exchange VPN route information with other PEs via BGP after learning the local VPN route form CE. It only maintains the VPN route directly connected with it rather than all VPN routes in the service provider network.

P router only maintains routes to PE, without learning any VPN route information.

Then transmitting VPN traffic in the MPLS backbone network, the ingress PE serves as the Ingress LSR (Label Switch Router), the egress PE the Egress LSR, and P router the Transit LSR.

3.1.2 Basic Concept of BGP/MPLS VPN

Site

"Site" is a concept usually mentioned when introducing VPN, which can be understood from the following aspects:

- Site is a set of IP systems with IP connectivity between each other. This connectivity is independent of SP network.
- The division of site is based on the topology of devices instead of devices' location, although in most cases, the devices in a site locate next to each other.
- The devices in a site can belong to multiple VPN. In other words, a site can belong to multiple VPN;
- Site connects to SP network via CE. One site can include multiple CE while a CE can only belong to one site.

Multiple sites connected to the same SP network can be divided into different sets according to special policies, which only allow intercommunication via the SP network to

happen between the sites within the same set. Such sets are VPN.

VRF

VRF (VPN Routing & Forwarding Instance), consisting of VPN IP route table and VPN IP forwarding table (the forwarding table contains the MPLS encapsulation information), is the core entry of MPLS VPN packet forwarding. Each VPN has its own independent VRF. The VRF address spaces of different VPN can overlap with each other. A PE/P router in the MPLS VPN network usually contains multiple independent VRF.

Overlapping Address Space

VPN is a private network, which means each VPN manages its own address range independently. This range is called Address Space.

The address spaces of different VPN may partially overlap with each other. For example, if VPN1 and VPN2 both use the segment of 10.110.10.0/24, there would be Overlapping Address Space.

VPN instance

In the MPLS VPN, the route isolation between different VPN is implemented via VPN instance.

PE creates and maintains a special VPN instance for every site directly connected to it. VPN site contains the VPN membership and route rules of the corresponding site. If the customers of a site belong to more than one VPN, then its VPN instance will contain the information of all those VPN.

To guarantee the data independency and security of VPN, each VPN instance on PE has its own independent route table and IFIL (Label Forwarding Information Base).

To be specific, the information in VPN instances include: LFIB, IP route table, interfaces bound with VPN instance, and its management information (including RD, route filter policy, member interface list and etc).

VPN-IPv4 Address

The traditional BGP can't correctly handle the VPN routes with overlapping address spaces. Assume that VPN1 and VPN2 both use the segment of 10.110.10.0/24, and advertise separately a route reaching this segment, BGP will only choose one of the two routes, losing the one reaching the other VPN.

PE routers use MP-BPG to advertise VPN routes between each other and solve the above problem via VPN-IPv4 address family.

A VPN-IPv4 address consists of 12 bytes, including 8 bytes of RD (Route Distinguisher) and 4 bytes of IPv4 address prefix.

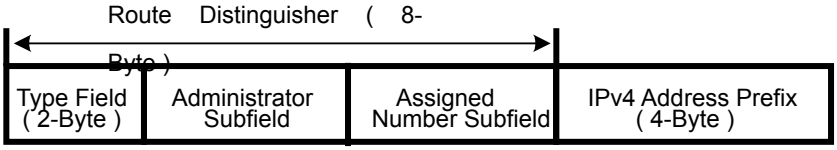

Fig 2-2 VPN-IPv4 Address Structure

After receiving the regular IPv4 routes from CE, PE should advertise these private

network VPN routes to the remote PE. The independency of the private network routes is based on the additional RD patched to them.

SP can independently distribute globally unique RD, thus, even the VPN from different SP networks use the same IPv4 address space, the PE routers can advertise different routes to them.

It is recommended to allocate a special RD for each VPN instance on the PE to ensure all routes reaching the same CE uses the same RD. the VPN-IPv4 address whose RD is 0 is a globally unique IPv4 address.

Adding RD is to a specific IPv4 prefix will make the latter globally unique, which is the meaning of RD.

RD may relate with ASN, in which case, it is a combination of an ASN and a random number; it may also relate with IP address, in which case, it is a combination of an IP address and a random number.

There are two RD formats, differing with each other via 2 bytes of Type filed:

- ☞ If Type is 0, the Administrator sub-field takes up 2 bytes, Assigned Number sub-field takes up 4 bytes. The format would be: 16 bits of ASN: 32 bits of user-defined number. For example: 100:1
- ☞ If Type is 0, the Administrator sub-field takes up 2 bytes, Assigned Number sub-field takes up 4 bytes. The format would be: 32 bits of IPv4 address: 16 bits of user-defined number. For example: 172.1.1.1:1

To guarantee the global uniqueness of RD, please don't set the value of Administrator sub-filed as private ASN or private IP address.

VPN Target Attribute

BGP/MPLS VPN uses a 32 bit BGP extended community attribute – VPN Target (also called Route Target) to control the advertisement of VPN route information.

There are two types of VPN Target attribute used by VPN instances on PE routers:

- ☞ Export Target attribute: the local PE sets the Export Target attribute for the VPN-IPv4 routes it learns from the sites directly connected to it, before advertising the routes to other PE.
- ☞ Import Target Attribute: when receiving the VPN-IPv4 route advertised by other PE routers, PE will check their Export Target Attribute, and add the routes into corresponding VPN route table only when their Export Target attributes match the Import Target attributes of the VPN instances on it.

In other words, VPN Target attribute defines which sites can accept a VPN-IPv4 route, and a PE router can receive routes from witch sites.

Like RD, there are two VPN Target formats:

- ☞ 16 bits ASN : 32bits user-defined number, for example: 100:1
- ☞ 32bits IPv4 address: 16 bits user-defined number, for example: 172.1.1.1:1

MP-BGP

MP-BGP (Multiprotocol extensions for BGP-4) transmits VPN information and routes between PE routers. MP-BGP is backward-compatible, simultaneously supporting traditional IPv4 address family and other address family (such as VPN-IPv4 address family). It can ensure the advertisement of private network VPN routes only happens within the VPN, and can realize the communication between MPLS VPN members.

Routing Policy

On the basis of controlling VPN route advertisement via ingress and egress extended community, the import or export route policy can be used for a more precise control of importing and advertising VPN routes.

The import route policy can filter the routes importable for VPN instances according to the VPN target attribute of routes. It can deny the receipt of routes specified by the community in the import list. The export route policy can deny advertising the routes specified by the community in the export list.

After creating VNP instances, users can choose whether to configure import or export route policy.

Tunneling Policy

Tunneling Policy is used to choose tunnels for specified VPN instances messages.

Tunneling Policy is optional. After creating VNP instances, users can configure it. By default, it will choose LSP as the tunnel without load sharing (the load sharing number is 1). Besides, this policy only takes effect in one AS domain.

3.1.3 Forwarding BGP/MPLS VPN Messages

In basic L3VPN applications (not include Multi-AS VPN), the forwarding of VPN packets adopts the 2-layer label mode:

- The first layer (outer layer) labels will be switched within the backbone network, indicating a LSP from the PE to the remote PE. With this layer of label, VPN messages can reach the remote PE along the LSP.
- The second layer label (inner layer) will be used when the packet reaches CE from the remote PE, indicating which site to send the packet, or, more specifically, which CE it will reach. Thus the remote PE will find the correct interface to forward the packet according to this layer of label.

In some special conditions, two sites belonging to the same VPN may connect to a same PE, in which case, the only information matters is how to reach the remote CE.

The next figure demonstrates an example of forwarding VPN packets:

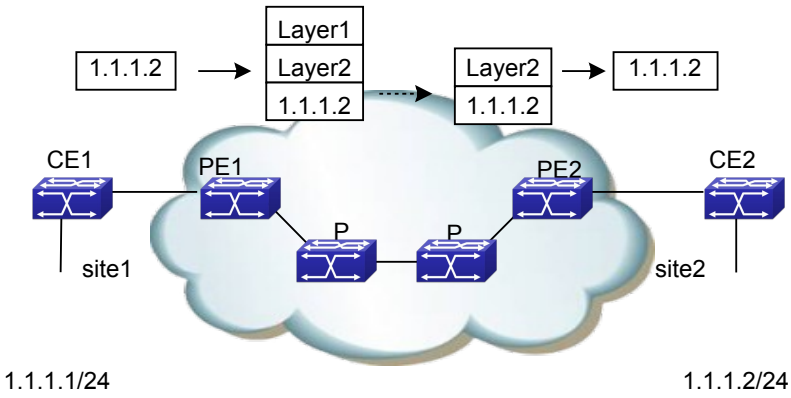

Fig 2-3 Forwarding VPN Packets

(1) Site1 sends an IP packet with a destination address of 1.1.1.2, which is sent by CE1 to PE1.
(2) PE1 looks up VPN-instance entries according to the interface receiving the packet and the destination address, then forwards the packet after adding two layers of label (inner and outer) to it, if there is a match.
(3) The MPLS network will send the packet to PE2 according to the outer layer label (removed when the packet reaching the last-hop of PE2, leaving only the inner layer) of it.
(4) PE1 looks up VPN-instance entries according to the inner layer of label and the destination address, then forwards the packet to CE2 after determining its egress interface.
(5) CE2 forwards the packet to its destination according to the regular IP forwarding process.

3.1.4 BGP/MPLS VPN Networking Resolution

In BGP/MPLS VPN networks, the advertisement and receipt of VPN routes between different sites are controlled by VPN Target Attribute. The configurations of VPN Export Target and Import Target are independent, both allowing multiple values, and hence can realize flexible VPN access control and various VPN networking resolutions.

Basic VPN

In the most basic instance, all users of a VPN form a closed user group, allowing the forwarding of traffic between them. But no user within the VPN can communicate with outside users.

In such networking, each VPN will obtain an exclusive VPN Target as its Export Target and Import Target, which should not be used by other VPN.

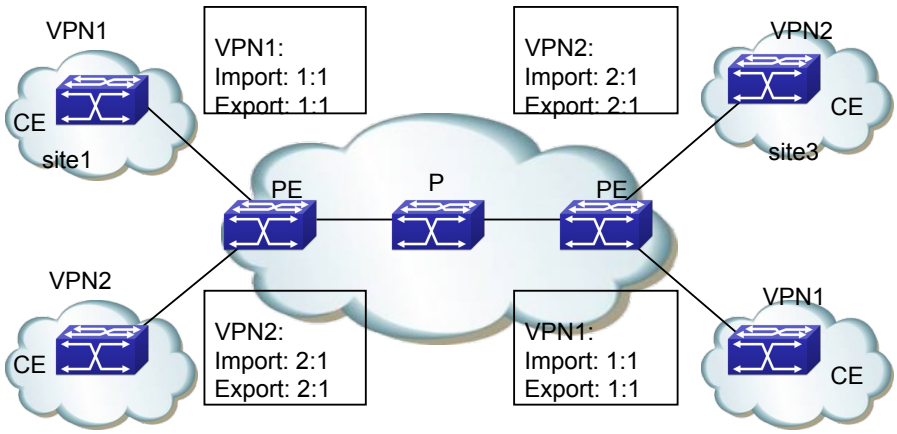

Fig 2-4 Basic VPN Networking Resolution

In the above figure, the VPN Target distributed by PE for VPN1 is 100:1; and that for VPN2 is 200:1. The sites of VPN1 can intercommunicate with each other, so do the two

of VPN2. But the intercommunication between sites in VPN1 and those in VPN2 arise forbidden.

Hub&Spoke VPN

To use a central access control device in VPN to control the intercommunication of other users, Hub&Spoke networking resolution is a good choice, so that the central device can monitor and filter the intercommunication between the devices at two ends.

Two VPN target is needed in this networking, one for “Hub”, the other for “Spoke”.

All sites should follow the following rules to configure VPN Target for VPN instances on PE:

- Spoke-PE: Export Target is “Spoke”, Import Target is “Hub”
- Hub-PE: two interfaces or sub-interfaces are needed, one for receiving routes from Spoke-PE, the Import Target of whose VPN instance is “Spoke”; the other for advertising routes to Spoke-PE, the Export Target of whose VPN instance is “Hub”.

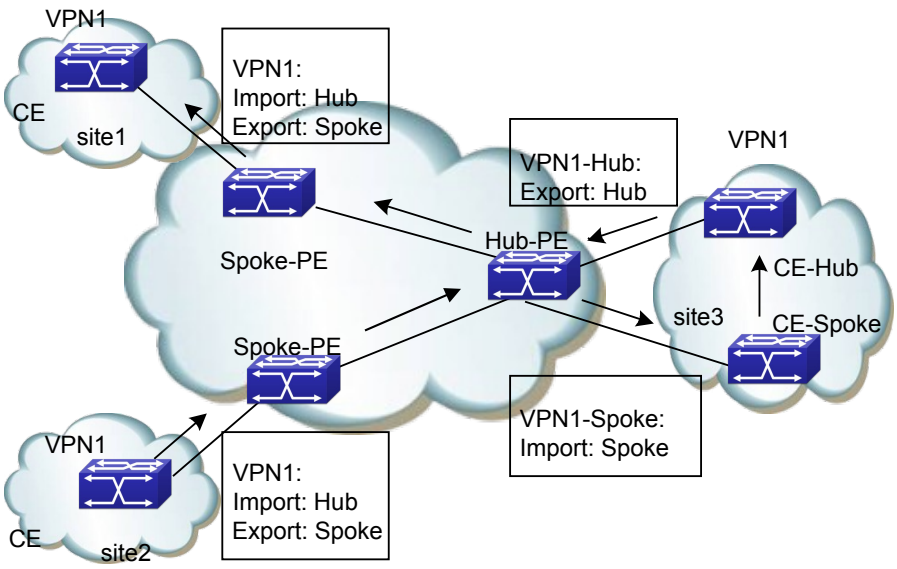

Fig 2-5 Hub&Spoke Networking Resolution

In the above figure, Spoke sites communicate with each other via Hub sites (the arrow in the figure is the route advertisement process from site2 to site1):

- Hub-PE can receive VPN-IPv4 routes advertised by all Spoke-PE
- The VPN-IPv4 routes advertised by Hub-PE can be received by all Spoke-PE;
- Since Hub-PE can advertise routes it learns from Spoke-PE to other Spoke-PE, the spoke sites can intercommunicate with each other via the Hub site.
- The Import Target attribute of any Spoke-PE is different from the Export Target attribute of other Spoke-PE. So, any pair of Spoke-PE cannot advertise VPN-IPv4 routes to each other or intercommunicate directly.

Extranet VPN

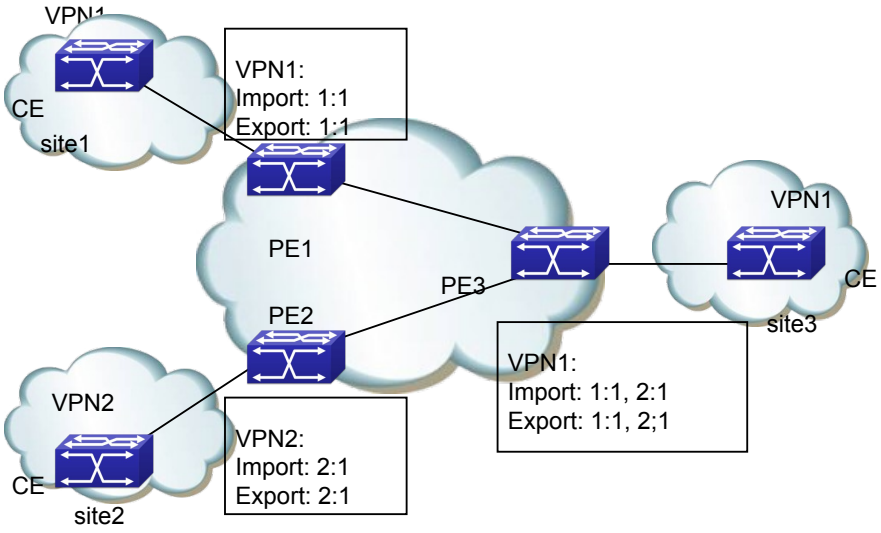

Fig 2-6 Extranet Networking Resolution

If a VPN user wants to provide some site resource of this VPN to outside users, the Extranet Networking resolution can solve the problem.

In this networking if a VPN needs to access the sharing site, its Export Target should be included in the Import Target of the sharing site VPN instances, and its Import Target should be included in the Export Target of the sharing site VPN instances.

In the above figure, site3 of VPN1 can be accessed by VPN1 and VPN2:

- PE3 can receive the VPN-IPv4 routes advertised by PE1 and PE2
- PE1 and PE2 can receive the VPN-IPv4 routes advertised by PE3
- Based on the above two conditions, site1 and site3 of VPN1 can intercommunicate, so do the site2 of VPN2 and site3 of VPN1.

PE3 won't advertise VPN-IPv4 routes from PE1 to PE2, or advertise the VPN-IPv4 route from PE2 to PE1 (the routes learnt from an IGBP neighbor won't be sent to other IBGP neighbors), so site1 of VPN1 and site2 of VPN2 can't intercommunicate.

3.1.5 BGP/MPLS VPN Route Advertisement

In basic BGP/MPLS VPN networks, VPN route advertisement concerns CE and PE, since P routers only maintains routes of the backbone network, and doesn't need any VPN route information. PE routers only maintain the VPN route information directly connected to it, not all VPN routes. SO the BGP/MPLS VPN network is easy to extend.

The VPN route advertisement process includes three parts to create a reachable route from the local CE to the remote CE, enabling the advertisement of VPN private network route information in the backbone network: from local CE to ingress PE, from the ingress PE to the egress PE, from egress PE to the remote CE.

The followings are introduction to the three parts:

The route information switch from the local CE to the ingress PE

CE will send the local VPN route to the PE directly connected to it after establishing

an adjacency to the latter.

CE can use static routes, RIP, OSPF, IS-IS or EBGP to send routes to PE, all in the form of standard IPv4 routes.

The route information switch from the ingress PE to the egress PE

PE will add RD and VPN target attributes to the VPN routes it learns from CE, then store these VPN-IPv4 routes into the VPN instances created for CE.

The ingress PE will advertise the VPN-IPv4 routes to the egress PE via MP-BGP. The egress PE will determine whether to add this route into the route table of VPN instance according to the routes' Export Target attribute and the import Target of the VPN instances it maintains.

Different PE ensure the intercommunication between them via IGP.

The route information switch between the egress PE to the remote CE

Like the route information switch from the local CE to the ingress PE, there are many available methods for the remote CE to learn VPN routes the egress PE, including static route, RIP, OSPF, IS-IS and EBGP.

3.1.6 Multi-AS VPN Introduction

In real networking applications, multiple sites of a user VPN may connect to SP with different ASN, or to different AS of the same SP. Such applications of one VPN crossing multiple autonomy systems are called Multi-AS VPN. RFC 2547 provides three Multi-AS VPN resolutions:

- ☞ VRF-to-VRF：ASBR use VRF interface to create EBGP neighbors and manage VPN routes, which is also called Inter-Provider Option A;
- ☞ EBGP Redistribution of labeled VPN-IPv4 routes： ASBR use MP-EBGP to advertise label VPN-IPv4 routes, which is also called Inter-Provider Option B;
- ☞ Multihop EBGP redistribution of labeled VPN-IPv4 routes：PE use Multi-hop MP-EBGP to advertise label VPN-IPv4 routes, which is also called Inter-Provider Option C.

At present we support the first resolution: VRF-to-VRF Multi-VPN resolution.

Multi-VPN resolution

As demonstrated in the nest figure, in this mode, PE routers from two AS directly connects with each other, and serve as ASBR of the AS they belong to. These PE routers (ASBR) connect with each other via VRF interfaces, import all RT this system need the other end to learn, export all RT this system want to obtain from the other, and establish EBGP connections through the VRF interfaces. As a result, the CE they serve will be able to intercommunicate with and isolate from each other like locating in the same AS, with two PE routers treating each other as their own CE. Packets will be forwarded within the AS as VPN packets in the 2-layer label mode, and forwarded as regular IP packets between ASBR.

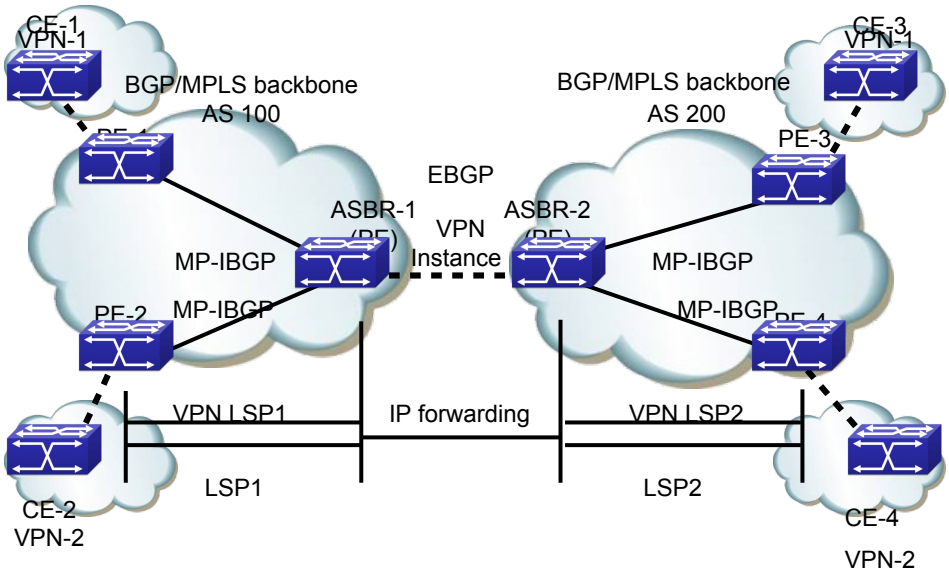

Fig 2-7 Multi-AS VPN Networking

- ☞ The advantage of this Multi-AS VPN mode is easy to realize: no special configuration is needed between the two PE serving as ASBR.
- ☞ The disadvantage is poor extendibility: the PE serving as ASBR need to manage all VPN routes, and create VPN instances for each VPN. This will cause too many VPN-IPv4 routes on PE.

3.2 BGP MPLS VPN Configuration

BGP MPLS VPN configuration task sequence:

1. Enable globally MPLS (necessary)
2. Configure VPN instances (necessary)
   (1) Create VPN instances, and enter the VPN instance view.
   (2) RD Configure the VPN instance RD
   (3) Configure the VPN instance RT
   (4) Configure the VPN instance to relate with the interface
3. Configure basic MPLS VPN (necessary)
   (1) Configure to use EBGP between PE-CE
      1) Configure the remote PE as the public network VPNv4 neighbor
      2) Enter the BGP-VPN instance view
      3) Configure CE as the VPN private network neighbor
      4) Advertise local private network routes
   (2) Configure to use EBGP between PE-CE
      1) Configure the remote PE as the public network VPNv4 neighbor
      2) Create the OSPF instance between PE-CE, and enter the Router OSPF view.

3) Enable OSPF in the segment between PE-CE
4) Configure to re-advertise BGP routes
5) Enter the BGP-VPN instance view
6) Configure to re-advertise OSPF routes
7) Advertise local private network routes

(3) RIP Configure to use EBGP between PE-CE
1) Configure the remote PE as the public network VPNv4 neighbor
2) Enter the RIP VPN instance view
3) Enable RIP in the segment between PE-CE
4) Configure to re-advertise BGP routes
5) Enter the BGP-VPN instance view
6) Configure to re-advertise RIP routes
7) Advertise local private network routes

(4) Configure to use static routes between PE-CE
1) Configure the remote PE as the public network VPNv4 neighbor
2) Configure static VPN routes
3) Enter the BGP-VPN instance view
4) Configure to re-advertise static routes
5) Advertise local private network routes

1. Enable MPLS (necessary)

Command	Explanation
Global Configuration Mode	
mpls enable **no mpls enable**	Necessary Enable MPLS; the no operation will disable MPLS.

2. Configure VPN instances (necessary)

(1) Create VPN instances and enter VPN instance view
(2) Configure VPN instance RD
(3) Configure VPN instance RT
(4) Configure VPN instance to relate with the interface

Command	Explanation
Global Configuration Mode	
[no] ip vrf <*vrf-name*>	Necessary Create VPN instances; no VPN instance is created by default.
VRF Configuration Mode	
[no] rd <*ASN:nn*_or_*IP-address:nn*>	Necessary Configure VPN instance RD; no RD is created by default.
[no] route-target {import | export | both} <*rt-value*>	Necessary Configure VPN instance RT.

Interface Configuration Mode	
[no] ip vrf forwarding < ***vrf-name*** >	Necessary Configure VPN instance to relate with the interface.
mpls proxy loopback-group <1-max_agg_num> **no mpls proxy loopback-group**	Enable MPLS proxy, the default does not enable the function, enable MPLS proxy when the boardcard of VRF ingress does not support MPLS.
[no] ip address <*ip-address*> <*mask*>	Necessary Configure the private network IP address of the interface directly connecting PE and CE.

3 Configure basic MPLS VPN (necessary)

(1) Configure to use EBGP between PE-CE

1) Configure the remote PE as the public network VPNv4 neighbor
2) Enter the BGP-VPN instance view
3) Configure CE as the VPN private network neighbor
4) Advertise local private network routes

Command	Explanation
BGP Protocol Configuration Mode	
neighbor <*ip-address*> remote-as <*as-num*>	necessary Configure the remote PE as the public network VPNv4 neighbor. It's suggest to select loopback interface to set up the BGP neighbor among public network PE.
neighbor <*ip-address*> update-source <*as-num*>	Point the local loopback interface for set up neighbor.
Enter the BGP-VPNv4 view	
address-family vpnv4 [unicast]	necessary Create BGP VPNv4. No VPNv4 is created by default.
[no] neighbor <*ip-address*> active	optional Activate all neighbors in VPNv4. All neighbors in VPNv4 view are inactive by default.
BGP Protocol Configuration Mode	
[no] address-family ipv4 {unicast| multicast|vrf <*vrf-nam*>}	optional; Create BGP protocol IPv4 and enter the BGP-VPN instance view. No IPv4 is created by default.
BGP-VPN instance view	

[no] neighbor <*ip-address*> remote-as <*as-num*>	optional Configure CE as the VPN private network neighbor. No private network neighbor is configured by default.
[no] neighbor <*ip-address*> active	optional Activate all neighbors in VPNv4. All neighbors in VPNv4 view are active by default.
[no] redistribute {connected|ospf| rip| static}	optional Configure to re-advertise the directly connected routes and other protocol routes. No re-advertisement of any route by default.

(2) Configure to use EBGP between PE-CE

1) Configure the remote PE as the public network VPNv4 neighbor
2) Create the OSPF instance between PE-CE, and enter the Router OSPF view
3) Enable OSPF in the segment between PE-CE
4) Configure to re-advertise BGP routes
5) Enter the BGP-VPN instance view
6) Configure to re-advertise OSPF routes
7) Advertise local private network routes

Command	Explanation
BGP Protocol Configuration Mode	
neighbor <*ip-address*> remote-as <*as-num*>	necessary Configure the remote PE as the public network VPNv4 neighbor. It's suggest to select loopback interface to set up the BGP neighbor among public network PE.
neighbor <*ip-address*> update-source <*as-num*>	Point the local loopback interface for set up neighbor.
Enter the BGP-VPNv4 view	
address-family vpnv4 [unicast]	necessary Create BGP VPNv4. No VPNv4 is created by default.
[no] neighbor <*ip-address*> active	optional Activate all neighbors in VPNv4. All neighbors in VPNv4 view are active by default.
Global Configuration Mode	
[no] router ospf [<*process_id*> [<*vrf-nam*>]]	optional Create the OSPF instance between PE-CE, and enter the Router OSPF view.

OSPF VPN instance view	
[no] network {<network> <mask> | <network>/<prefix>} area <area_id>	optioanal Enable OSPF in the segment between PE-CE. Enabled in no segment by default.
[no] redistribute { bgp | connected | static| rip | kernel} [metric-type {1 | 2}] [tag <tag>] [metric <cost_value>] [router-map <WORD>]	optional Configure to re-advertise the BGP routes. No re-advertisement of any route by default.
BGP Protocol Configuration Mode	
[no] address-family ipv4 {unicast| multicast|vrf <*vrf-nam*>}	optional create BGP VPNv4 and enter the BGP-VPN instance view. No VPNv4 is created by default.
BGP-VPN instance view	
[no] redistribute {connected | ospf | rip | static}	optional Configure to re-advertise the directly connected routes and other protocol routes. No re-advertisement of any route by default.

(3) Configure to use EBGP between PE-CE

1) Configure the remote PE as the public network VPNv4 neighbor
2) Enter the RIP VPN instance view
3) Enable RIP in the segment between PE-CE
4) Configure to re-advertise BGP routes
5) Enter the BGP-VPN instance view
6) Configure to re-advertise RIP routes
7) Advertise local private network routes

Command	Explanation
BGP Protocol Configuration Mode	
neighbor <*ip-address*> remote-as <*as-num*>	necessary Configure the remote PE as the public network VPNv4 neighbor. It's suggest to select loopback interface to set up the BGP neighbor among public network PE.
neighbor <*ip-address*> update-source <*as-num*>	Point the local loopback interface for set up neighbor.
Enter the BGP-VPNv4 view	
address-family vpnv4 [unicast]	necessary Create BGP VPNv4. No VPNv4 is created by default.

[no] neighbor <*ip-address*> active	optional Activate all neighbors in VPNv4. All neighbors in VPNv4 view are active by default.
RIP Protocol Configuration Mode	
[no] address-family ipv4 vrf <*vrf-name*>	optional Create RIP IPv4 protocol family and enter RIP VPN instance view.
RIP VPN instance view	
[no] network {*A.B.C.D/M*|*ifname*| *vlan* <*id*>|loopback <*1-1024*> }	optional Enable the RIP between PE-CE.
[no] redistribute { kernel | connected | static | ospf | isis | bgp} [metric <value>] [route-map<word>]	optional Configure to re-advertise the BGP routes. No re-advertisement of any route by default.
BGP Protocol Configuration Mode	
[no] address-family ipv4 {unicast| multicast|vrf <*vrf-nam*>}	optional Create BGP VPNv4 and enter the BGP-VPN instance view. No VPNv4 is created by default.
BGP-VPN instance view	
[no] redistribute {connected | ospf | rip | static}	optional Configure to re-advertise the directly connected routes and other protocol routes. No re-advertisement of any route by default.

(4) Configure to use static routes between PE-CE

1) Configure the remote PE as the public network VPNv4 neighbor
2) Configure static VPN routes
3) Enter the BGP-VPN instance view
4) Configure to re-advertise static routes
5) Advertise local private network routes

Command	Explanation
BGP Protocol Configuration Mode	
neighbor <*ip-address*> remote-as <*as-num*>	necessary Configure the remote PE as the public network VPNv4 neighbor. It's suggest to select loopback interface to set up the BGP neighbor among public network PE.
neighbor <*ip-address*> update-source <*as-num*>	Point the local loopback interface for set up neighbor.
Enter the BGP-VPNv4 view	

address-family vpnv4 [unicast]	necessary Create BGP VPNv4. No VPNv4 is created by default.
[no] neighbor <*ip-address*> active	optional Activate all neighbors in VPNv4. All neighbors in VPNv4 view are active by default.
Global Configuration Mode	
[no] ip route vrf <vrf-name> {<ip-prefix> <mask>|<ip-prefix/<prefix-length>} {<gateway-address>|null0}	optional Manually configure the static VPN routes between PE-CE.
BGP Protocol Configuration Mode	
[no] address-family ipv4 {unicast| multicast|vrf <*vrf-nam*>}	optional Create BGP VPNv4 and enter the BGP-VPN instance view. No VPNv4 is created by default.
BGP-VPN instance view	
[no] redistribute {connected|ospf| rip| static}	optional Configure to re-advertise the static routes, directly connected routes and other protocol routes. No re-advertisement of any route by default.

3.3 BGP MPLS VPN Typical Instances

3.3.1 Create BGP MPLS VPN between PE-CE via EBGP

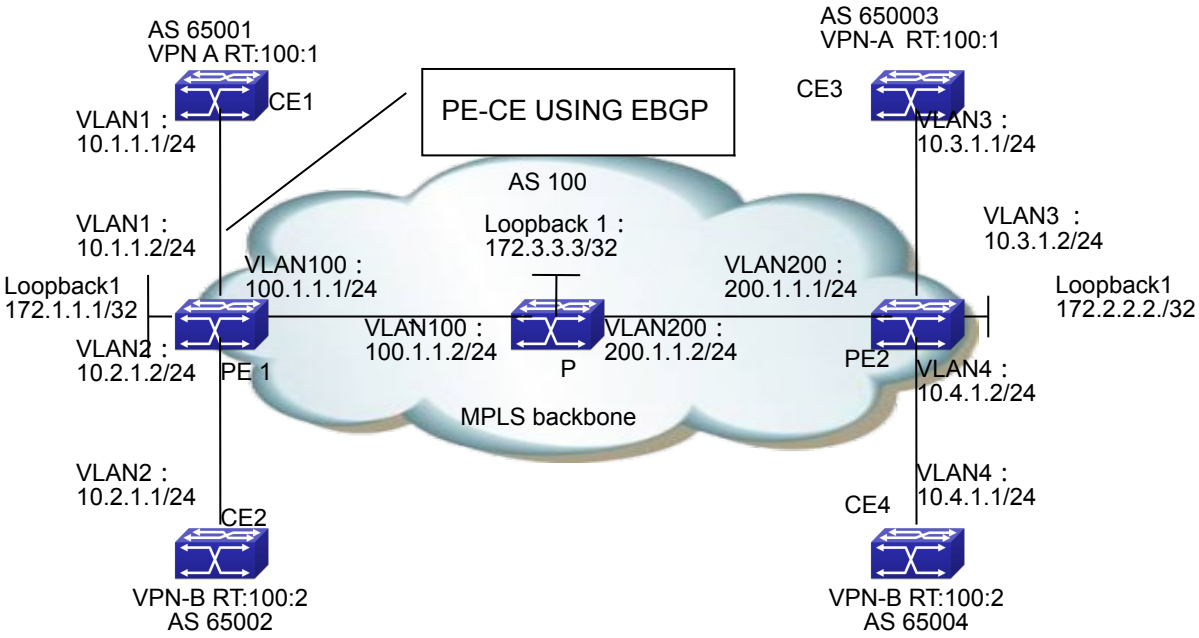

Fig 2-8 Create BGP MPLS VPN between PE-CE via EBGP

The configuration of CE1 is as follows : (the configurations of CE2~CE4 are similar)

CE1#config
CE1(config)# interface vlan 1
CE1(config-if-Vlan1)#ip address 10.1.1.1 255.255.255.0
CE1(config-if-Vlan1)#exit
CE1(config)# router bgp 65001
CE1(config-router)#neighbor 10.1.1.2 remote-as 100
CE1(config-router)#redistribute connect
CE1(config-router)#exit

The configuration of MPLS BGP on switch PE1 is as follows:

(1) Configure VPN instances
PE1#config
PE1(config)#ip vrf vpna
PE1(config-vrf)#rd 100:1
PE1(config-vrf)#route-target both 100:1
PE1(config)#ip vrf vpnb
PE1(config-vrf)#rd 100:2
PE1(config-vrf)#route-target both 100:2
(2) Configure to bind the interface with the VPN instances
PE1(config)# interface vlan 1
PE1(config-if-Vlan1)# ip vrf forwarding vpna

PE1(config-if-Vlan1)#ip address 10.1.1.2 255.255.255.0
PE1(config-if-Vlan1)#exit
PE1(config)# interface vlan 2
PE1(config-if-Vlan2)# ip vrf forwarding vpnb
PE1(config-if-Vlan2)#ip address 10.2.1.2 255.255.255.0
PE1(config-if-Vlan2)#exit
(3) Globally enable MPLS and LDP
PE1(config)#mpls enable
PE1(config)#router ldp
PE1(config-router)#exit
(4) LDP Configure the interface and enable LDP
PE1(config)# interface loopback 1
PE1(config-if-Loopback1)# ip address 172.1.1.1 255.255.255.255
PE1(config-if-Loopback1)# exit
PE1(config)# interface vlan 100
PE1(config-if-Vlan100)#ip address 100.1.1.1 255.255.255.0
PE1(config-if-Vlan100)#label-switching
PE1(config-if-Vlan100) #ldp enable
PE1(config-if-Vlan100)#exit
(5) Enable OSPF to advertise the inner network routes
PE1(config)#router ospf
PE1(config-router)# ospf router-id 172.1.1.1
PE1(config-router)# network 0.0.0.0/0 area 0
PE1(config-router)# redistribute connected
(6) Configure BGP
PE1(config)# router bgp 100
PE1(config-router)#neighbor 172.2.2.2 remote-as 100
PE1(config-router)#neighbor 172.2.2.2 update-source 172.1.1.1
PE1(config-router)#address-family vpnv4
PE1(config-router-af)#neighbor 172.2.2.2 activate
PE1(config-router-af)#exit
PE1(config-router)# address-family ipv4 vrf vpna
PE1(config-router-af)#neighbor 10.1.1.1 remote-as 65001
PE1(config-router-af)#redistribute connected
PE1(config-router-af)#exit
PE1(config-router)# address-family ipv4 vrf vpnb\
PE1(config-router-af)#neighbor 10.2.1.1 remote-as 65002
PE1(config-router-af)#redistribute connected
PE1(config-router-af)#exit
PE1(config-router)#exit

The configuration of router P is as follows:

(1) Globally enable MPLS and configure LDP on related interfaces.
P#config
P(config)#mpls enable

P(config)#router ldp
P(config-router)#exit
P(config)# interface loopback 1
P(config-if-Loopback1)# ip address 172.3.3.3 255.255.255.255
P(config-if-Loopback1)# exit
P(config)#interface vlan 100
P(config-if-Vlan100)#ip address 100.1.1.2 255.255.255.0
P(config-if-Vlan100)#label-switching
P(config-if-Vlan100)#ldp enable
P(config-if-Vlan100)#exit
P(config)#interface vlan200
P(config-if-Vlan200)#ip address 200.1.1.2 255.255.255.0
P(config-if-Vlan200)#label-switching
P(config-if-Vlan200)#ldp enable
P(config-if-Vlan200)#exit
(2) Configure OSPF
P(config)#router ospf
P(config-router)# ospf router-id 172.3.3.3
P(config-router)# network 0.0.0.0/0 area 0
P(config-router)# redistribute connected

The configuration of switch PE2 is as follows:

(1) Configure VPN instances
PE2#config
PE2(config)#ip vrf vpna
PE2(config-vrf)#rd 100:1
PE2(config-vrf)#route-target both 100:1
PE2(config)#ip vrf vpnb
PE2(config-vrf)#rd 100:2
PE2(config-vrf)#route-target both 100:2
(2) Configure to bind the interface with the VPN instances
PE2(config)# interface vlan 3
PE2(config-if-Vlan3)# ip vrf forwarding vpna
PE2(config-if-Vlan3)#ip address 10.3.1.2 255.255.255.0
PE2(config-if-Vlan3)#exit
PE2(config)# interface vlan 4
PE2(config-if-Vlan4)# ip vrf forwarding vpnb
PE2(config-if-Vlan4)#ip address 10.4.1.2 255.255.255.0
PE2(config-if-Vlan4)#exit
(3) Globally enable MPLS and LDP
PE2(config)#mpls enable
PE1(config)#router ldp
PE1(config-router)#exit
(4) LDP Configure the interface and enable LDP
PE2(config)# interface loopback 1

PE2(config-if-Loopback1)# ip address 172.2.2.2 255.255.255.255
PE2(config-if-Loopback1)# exit
PE2(config)# interface vlan 200
PE2(config-if-Vlan200)#ip address 200.1.1.1 255.255.255.0
PE2(config-if-Vlan200)#label-switching
PE2(config-if-Vlan200) #ldp enable
PE2(config-if-Vlan200)#exit
(5) Enable OSPF to advertise the inner network routes
PE2(config)#router ospf
PE2(config-router)# ospf router-id 172.2.2.2
PE2(config-router)# network 0.0.0.0/0 area 0
PE2(config-router)# redistribute connected
(6) Configure BGP
PE2(config)# router bgp 100
PE2(config-router)#neighbor 172.1.1.1 remote-as 100
PE2(config-router)#neighbor 172.1.1.1 update-source 172.2.2.2
PE2(config-router)#address-family vpnv4
PE2(config-router-af)#neighbor 172.1.1.1 activate
PE2(config-router-af)#exit
PE2(config-router)# address-family ipv4 vrf vpna
PE2(config-router-af)#neighbor 10.3.1.1 remote-as 65003
PE2(config-router-af)#redistribute connected
PE2(config-router-af)#exit
PE2(config-router)# address-family ipv4 vrf vpnb
PE2(config-router-af)#neighbor 10.4.1.1 remote-as 65004
PE2(config-router-af)#redistribute connected
PE2(config-router-af)#exit
PE2(config-router)#exit

3.3.2 Create BGP MPLS VPN between PE-CE via OSPF

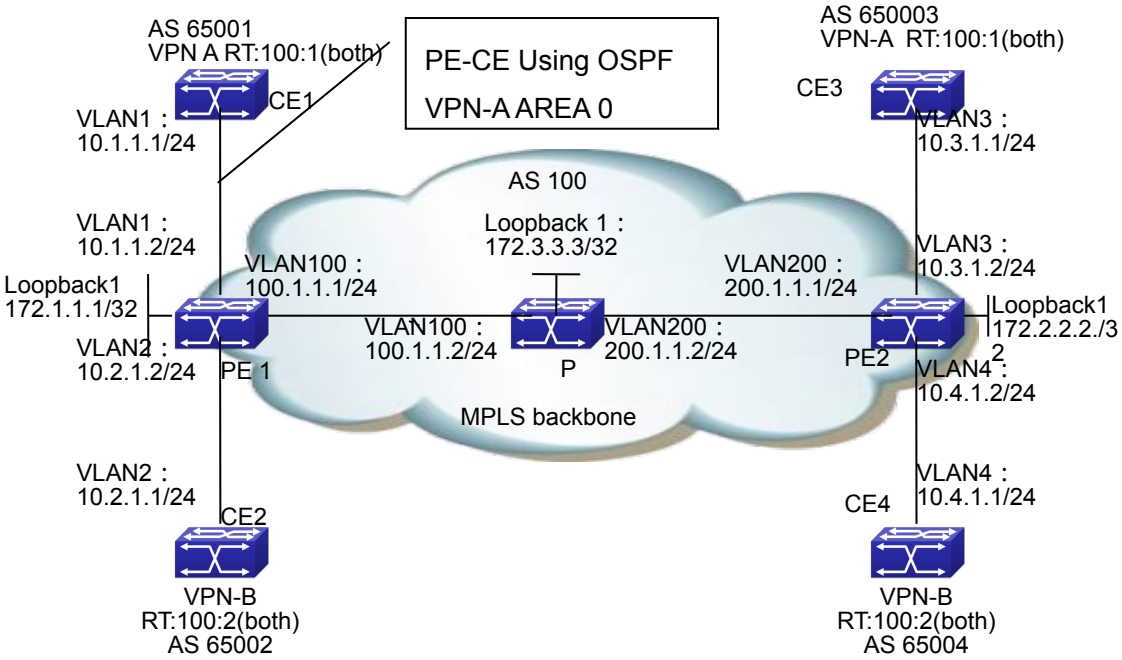

Fig 2-9 Create BGP MPLS VPN between PE-CE via OSPF

The configuration of CE1 is as follows : (the configurations of CE2~CE4 are similar)

```
CE1#config
CE1(config)# interface vlan 1
CE1(config-if-Vlan1)#ip address 10.1.1.1 255.255.255.0
CE1(config-if-Vlan1)#exit
CE1(config)# router ospf
CE1(config-router)#network 0.0.0.0/0 area 0
CE1(config-router)#redistribute connect
CE1(config-router)#exit
```

The confiugraiton of MPLS BGP on switch PE1 is as follows : (the configuration of PE2 is similar)

(1) Configure VPN instances

```
PE1#config
PE1(config)#ip vrf vpna
PE1(config-vrf)#rd 100:1
PE1(config-vrf)#route-target both 100:1
PE1(config)#ip vrf vpnb
PE1(config-vrf)#rd 100:2
PE1(config-vrf)#route-target both 100:2
```

(2) Configure to bind the interface with the VPN instances

```
PE1(config)# interface vlan 1
```

PE1(config-if-Vlan1)# ip vrf forwarding vpna
PE1(config-if-Vlan1)#ip address 10.1.1.2 255.255.255.0
PE1(config-if-Vlan1)#exit
PE1(config)# interface vlan 2
PE1(config-if-Vlan2)# ip vrf forwarding vpnb
PE1(config-if-Vlan1)#ip address 10.2.1.2 255.255.255.0
PE1(config-if-Vlan1)#exit
(3) Globally enable MPLS and LDP
PE1(config)#mpls enable
PE1(config)#router ldp
PE1(config-router)#exit
(4) LDP Configure the interface and enable LDP
PE1(config)# interface loopback 1
PE1(config-if-Loopback1)# ip address 172.1.1.1 255.255.255.255
PE1(config-if-Loopback1)# exit
PE1(config)# interface vlan 100
PE1(config-if-Vlan100)#ip address 100.1.1.1 255.255.255.0
PE1(config-if-Vlan100)# label-switching
PE1(config-if-Vlan100) #ldp enable
PE1(config-if-Vlan100)#exit
(5) Enable OSPF to advertise the inner network routes
PE1(config)#router ospf
PE1(config-router)# ospf router-id 172.1.1.1
PE1(config-router)# network 0.0.0.0/0 area 0
PE1(config-router)# redistribute connected
PE1(config-router)#exit
(6) Enable OSPF VRF to advertise the private network routes
PE1(config)#router ospf 1 vpna
PE1(config-router)# network 0.0.0.0/0 area 0
PE1(config-router)#redistribute connected
PE1(config-router)#redistribute bgp
PE1(config-router)#exit
PE1(config)#router ospf 1 vpnb
PE1(config-router)# network 0.0.0.0/0 area 0
PE1(config-router)#redistribute connected
PE1(config-router)#redistribute bgp
PE1(config-router)#exit
(7) Configure BGP
PE1(config)# router bgp 100
PE1(config-router)#neighbor 172.2.2.2 remote-as 100
PE1(config-router)#neighbor 172.2.2.2 update-source 172.1.1.1
PE1(config-router)#address-family vpnv4
PE1(config-router-af)#neighbor 172.2.2.2 activate
PE1(config-router-af)#exit

```
PE1(config-router)# address-family ipv4 vrf vpna
PE1(config-router-af)#redistribute connected
PE1(config-router-af)#redistribute ospf
PE1(config-router-af)#exit
PE1(config-router)# address-family ipv4 vrf vpnb
PE1(config-router-af)#redistribute connected
PE1(config-router-af)#redistribute ospf
PE1(config-router-af)#exit
PE1(config-router)#exit
```

The configuration of router P is as follows:

(1) Globally enable MPLS and configure LDP on related interfaces.

```
P#config
P(config)#mpls enable
P(config)#router ldp
P(config-router)#exit
P(config)# interface loopback 1
P(config-if-Loopback1)# ip address 172.3.3.3 255.255.255.255
P(config-if-Loopback1)# exit
P(config)#interface vlan 100
P(config-if-Vlan100)#ip address 100.1.1.2 255.255.255.0
P(config-if-Vlan100)#label-switching
P(config-if-Vlan100)#ldp enable
P(config-if-Vlan100)#exit
P(config)#interface vlan200
P(config-if-Vlan200)#ip address 200.1.1.2 255.255.255.0
P(config-if-Vlan200)#label-switching
P(config-if-Vlan200)#ldp enable
P(config-if-Vlan200)#exit
```

(2) Configure OSPF

```
P(config)#router ospf
P(config-router)# ospf router-id 172.3.3.3
P(config-router)# network 0.0.0.0/0 area 0
P(config-router)# redistribute connected
```

3.3.3 Create BGP MPLS VPN between PE-CE via RIP

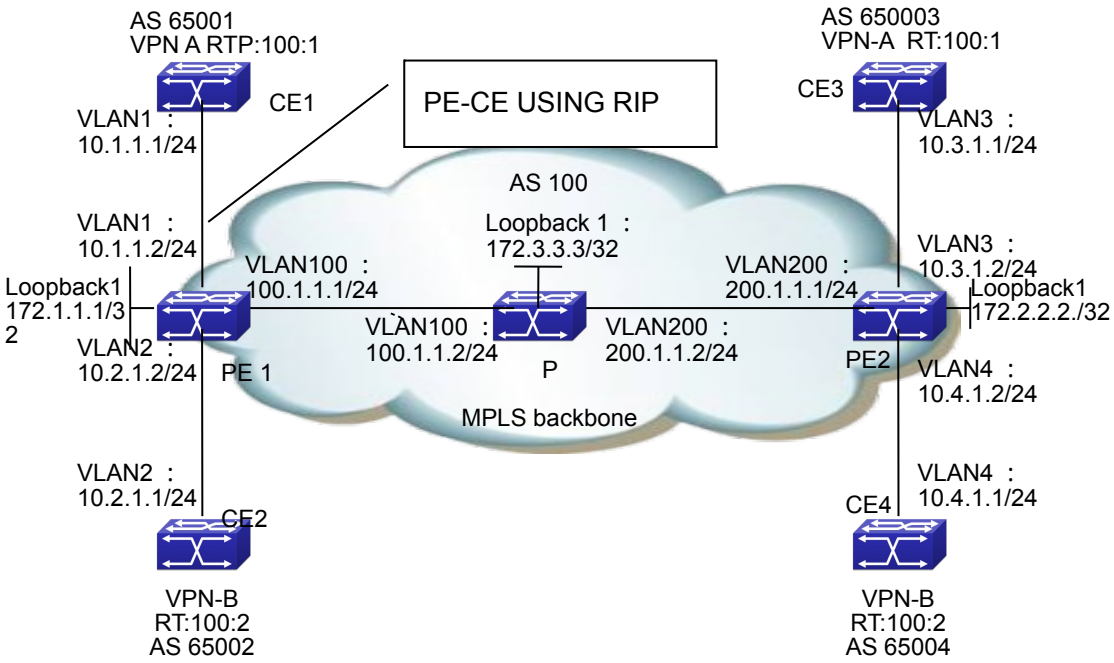

Fig 2-10 Create BGP MPLS VPN between PE-CE via RIP

The configuration of CE1 is as follows : (the configurations of CE2~CE4 are similar)

CE1#config
CE1(config)# interface vlan 1
CE1(config-if-Vlan1)#ip address 10.1.1.1 255.255.255.0
CE1(config-if-Vlan1)#exit
CE1(config)# router rip
CE1(config-router)#network 0.0.0.0/0
CE1(config-router)#redistribute connect
CE1(config-router)#exit

The confiugraiton of MPLS BGP on switch PE1 is as follows : (the configuration of PE2 is similar)

(1) Configure VPN instances
PE1#config
PE1(config)#ip vrf vpna
PE1(config-vrf)#rd 100:1
PE1(config-vrf)#route-target both 100:1
PE1(config)#ip vrf vpnb
PE1(config-vrf)#rd 100:2
PE1(config-vrf)#route-target both 100:2
(2) Configure to bind the interface with the VPN instances
PE1(config)# interface vlan 1
PE1(config-if-Vlan1)# ip vrf forwarding vpna
PE1(config-if-Vlan1)#ip address 10.1.1.2 255.255.255.0

PE1(config-if-Vlan1)#exit
PE1(config)# interface vlan 2
PE1(config-if-Vlan2)# ip vrf forwarding vpnb
PE1(config-if-Vlan1)#ip address 10.2.1.2 255.255.255.0
PE1(config-if-Vlan1)#exit
(3) Globally enable MPLS and LDP
PE1(config)#mpls enable
PE1(config)#router ldp
PE1(config-router)#exit
(4) LDP Configure the interface and enable LDP
PE1(config)# interface loopback 1
PE1(config-if-Loopback1)# ip address 172.1.1.1 255.255.255.255
PE1(config-if-Loopback1)# exit
PE1(config)# interface vlan 100
PE1(config-if-Vlan100)#ip address 100.1.1.1 255.255.255.0
PE1(config-if-Vlan100)#label-switching
PE1(config-if-Vlan100) #ldp enable
PE1(config-if-Vlan100)#exit
(5) Enable OSPF to advertise the inner network routes
PE1(config)#router ospf
PE1(config-router)# ospf router-id 172.1.1.1
PE1(config-router)# network 0.0.0.0/0 area 0
PE1(config-router)# redistribute connected
PE1(config-router)#exit
(6) Enable OSPF VRF to advertise the private network routes
PE1(config)#router rip
PE1(config-router)#address-family ipv4 vrf vpna
PE1(config-router-af)#network 0.0.0.0/0
PE1(config-router-af)#redistribute connected
PE1(config-router-af)#redistribute bgp
PE1(config-router-af)#exit
PE1(config-router)#address-family ipv4 vrf vpnb
PE1(config-router-af)#network 0.0.0.0/0
PE1(config-router-af)#redistribute connected
PE1(config-router-af)#redistribute bgp
PE1(config-router-af)#exit
PE1(config-router)#exit
(7) Configure BGP
PE1(config)# router bgp 100
PE1(config-router)#neighbor 172.2.2.2 remote-as 100
PE1(config-router)#neighbor 172.2.2.2 update-source 172.1.1.1
PE1(config-router)#address-family vpnv4
PE1(config-router-af)#neighbor 172.2.2.2 activate
PE1(config-router-af)#exit

PE1(config-router)# address-family ipv4 vrf vpna
PE1(config-router-af)#redistribute connected
PE1(config-router-af)#redistribute ospf
PE1(config-router-af)#exit
PE1(config-router)# address-family ipv4 vrf vpnb
PE1(config-router-af)#redistribute connected
PE1(config-router-af)#redistribute ospf
PE1(config-router-af)#exit
PE1(config-router)#exit

The configuration of switch P is as follows

(1) Globally enable MPLS and configure LDP on related interfaces.

P#config
P(config)#mpls enable
P(config)#router ldp
P(config-router)#exit
P(config)# interface loopback 1
P(config-if-Loopback1)# ip address 172.3.3.3 255.255.255.255
P(config-if-Loopback1)# exit
P(config)#interface vlan 100
P(config-if-Vlan100)#ip address 100.1.1.2 255.255.255.0
P(config-if-Vlan100)#label-switching
P(config-if-Vlan100)#ldp enable
P(config-if-Vlan100)#exit
P(config)#interface vlan200
P(config-if-Vlan200)#ip address 200.1.1.2 255.255.255.0
P(config-if-Vlan200)#label-switching
P(config-if-Vlan200)#ldp enable
P(config-if-Vlan200)#exit

(2) Configure OSPF

P(config)#router ospf
P(config-router)# ospf router-id 172.3.3.3
P(config-router)# network 0.0.0.0/0 area 0
P(config-router)# redistribute connected

3.3.4 Create BGP MPLS VPN between PE-CE via Static Routes

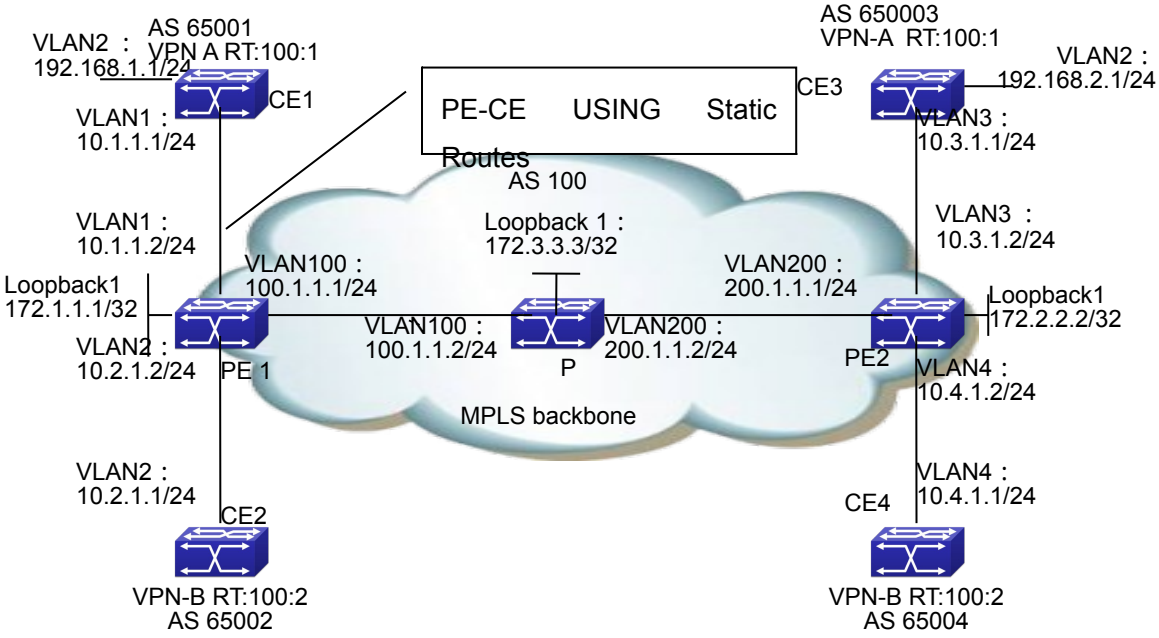

Fig 2-11 Create BGP MPLS VPN between PE-CE via Static Routes

The configuration of CE1 is as follows : (the configurations of CE2~CE4 are similar)

CE1#config
CE1(config)# interface vlan 1
CE1(config-if-Vlan1)#ip address 10.1.1.1 255.255.255.0
CE1(config-if-Vlan1)#exit
CE1(config)# interface loopback 1
CE1(config-if-Vlan1)#ip address 192.168.1.1 255.255.255.0
CE1(config-if-Vlan1)# exit
CE1(config)# ip route vrf vpna 192.168.2.1/24 10.1.1.2

The confiugraiton of MPLS BGP on switch PE1 is as follows : (the configuration of PE2 is similar)

(1) Configure VPN instances
PE1#config
PE1(config)#ip vrf vpna
PE1(config-vrf)#rd 100:1
PE1(config-vrf)#route-target both 100:1
PE1(config)#ip vrf vpnb
PE1(config-vrf)#rd 100:2
PE1(config-vrf)#route-target both 100:2
(2) Configure to bind the interface with the VPN instances
PE1(config)# interface vlan 1
PE1(config-if-Vlan1)# ip vrf forwarding vpna
PE1(config-if-Vlan1)#ip address 10.1.1.2 255.255.255.0

PE1(config-if-Vlan1)#exit
PE1(config)# interface vlan 2
PE1(config-if-Vlan2)# ip vrf forwarding vpnb
PE1(config-if-Vlan1)#ip address 10.2.1.2 255.255.255.0
PE1(config-if-Vlan1)#exit
(3) Globally enable MPLS and LDP
PE1(config)#mpls enable
PE1(config)#router ldp
PE1(config-router)#exit
(4) Configure the interface and enable LDP
PE1(config)# interface loopback 1
PE1(config-if-Loopback1)# ip address 172.1.1.1 255.255.255.255
PE1(config-if-Loopback1)# exit
PE1(config)# interface vlan 100
PE1(config-if-Vlan100)#ip address 100.1.1.1 255.255.255.0
PE1(config-if-Vlan100) #ldp enable
PE1(config-if-Vlan100)#exit
(5) Enable OSPF to advertise the inner network routes
PE1(config)#router ospf
PE1(config-router)# ospf router-id 172.1.1.1
PE1(config-router)# network 0.0.0.0/0 area 0
PE1(config-router)# redistribute connected
PE1(config-router)#exit
(6) Configure static private network routes
PE1(config)# ip route vrf vpna 192.168.1.1/24 10.1.1.2
PE1(config)# ip route vrf vpnb 192.168.2.1/24 10.1.1.2
PE1(config-router)#address-family ipv4 vrf vpna
PE1(config-router-af)#network 0.0.0.0/0
PE1(config-router-af)#redistribute connected
PE1(config-router)#exit
(7) Configure BGP
PE1(config)# router bgp 100
PE1(config-router)#neighbor 172.2.2.2 remote-as 100
PE1(config-router)#neighbor 172.2.2.2 update-source 172.1.1.1
PE1(config-router)#address-family vpnv4
PE1(config-router-af)#neighbor 172.2.2.2 activate
PE1(config-router-af)#exit
PE1(config-router)# address-family ipv4 vrf vpna
PE1(config-router-af)#redistribute connected
PE1(config-router-af)#redistribute static
PE1(config-router-af)#exit
PE1(config-router)# address-family ipv4 vrf vpnb
PE1(config-router-af)#redistribute connected
PE1(config-router-af)# redistribute static

PE1(config-router-af)#exit
PE1(config-router)#exit

The configuration of switch P is as follows

(1) Globally enable MPLS and configure LDP on related interfaces.

P#config
P(config)#mpls enable
P(config)#router ldp
P(config-router)#exit
P(config)# interface loopback 1
P(config-if-Loopback1)# ip address 172.3.3.3 255.255.255.255
P(config-if-Loopback1)# exit
P(config)#interface vlan 100
P(config-if-Vlan100)#ip address 100.1.1.2 255.255.255.0
P(config-if-Vlan100)#ldp enable
P(config-if-Vlan100)#exit
P(config)#interface vlan200
P(config-if-Vlan200)#ip address 200.1.1.2 255.255.255.0
P(config-if-Vlan100)#ldp enable
P(config-if-Vlan200)#exit

(2) Configure OSPF

P(config)#router ospf
P(config-router)# ospf router-id 172.3.3.3
P(config-router)# network 0.0.0.0/0 area 0
P(config-router)# redistribute connected

3.4 MPLS BGP VPN Troubleshooting

When configuring and using MPLS BGP VPN, some problems like incorrect physical connections, configuration errors may cause it to fail, so please pay attention to the following notices to avoid them:

- ☞ First, make sure the creation of OSPF neighbors between PE1, P and PE2, the advertisement of routes including the loopback interface and the creation of BGP neighbor between PE are correct.
- ☞ Second, make sure the LDP is globally enabled on PE1, P and PE2, and correctly enabled on active interfaces. Check whether the establishment of LDP sessions on PE1, P and PE2 is correct.
- ☞ Then, make sure the PE-CE route advertisement mode used when creating the VPN and corresponding configuration are correct. Check whether CE advertises related private network route to the remote PE. Please notice that CE needs no VRF instance. If EBGP is used to advertise the private network routes, the BGP ASN between CE1 and CE2 shouldn't be the same, or the loop detection of BGP will filter the corresponding private network routes.
- ☞ Next, make sure the BGP VPN instances on PE are correctly configured. When

using OSPF or RIP to create and advertise PE-CE routes, please import BGP routs and import corresponding OSPF and RIP routes to the BGP VPN instances. Implementing “show ip bgp vpnv4 all” on PE1 will display the route information of CE1 and CE2, if the configuration is correct. Implementing “show mpls vrf-table” on PE will display that the labels are distributed to corresponding private network routes, and the state is UP. If the Oper status in the vrf-table of the corresponding private network routes, use “show mpls ftn-table” to check whether the corresponding FEC create ftn.

- ☞ At last, if all above steps are correct, use “show ip route” on CE1 and CE2 to check the correct route information in the VPN. It is not recommended for users to create VPN via the static routes unless very familiar with BGP MPLS VPN.
- ☞ Besides, if no remote CE device can be checked on CE after saving the correction configuration and rebooting the device, please be patience, since the establishing OSPF, LDP, BGP connections and advertising routes are time-consuming.

Chapter 4 Public Network Access of MPLS VPN

4.1 Public Network Access Introduction

Public network access of VPN means the ability of VPN sites to access public Internet. RFC4364 defines the basic protocol regulations, including some methods for VPN to access Internet:

- ☞ Non-VRF Internet Access Mode
- ☞ VRF Internet Access Mode 1
- ☞ VRF Internet Access Mode 3

4.1.1 Non-VRF Internet Access Mode

As demonstrated in the next figure, in non-VRF Internet Access Mode, PE routers communicate with Internet gateways via non-VFP interface; and the Internet access traffic of VPN sites are forwarded according to the global route table of PE routers. The CE and PE routers capable of accessing Internet have two connections, one with the public network interface of PE (public network connection), the other with the priviate network interface of PE (private network connection). The global route table of PE routes can contain the whole or part of Internet routes, or only a default routes pointing to the Internet gateway. CE routers learn Interet routs via the public network connection, and advertise to PE via the public network connection the globally registered IP address sub-net routes in the VPN site, which will be advertised to the Internet gateway by PE and finally to Internet. The Internet access traffic of VPN sites is also sent and received by the public network connection. The private network connection between CE and PE is for the route learning of CE and advertising the private network routes in tehe VPN. The VPN sites also communicate via private network connections, and forward according to the VRF route table of PE routers. In this mode, the global and VRF rout table of PE routers are completely isolated ; and the distribution of VPN routes and Internet routes are completely independent.

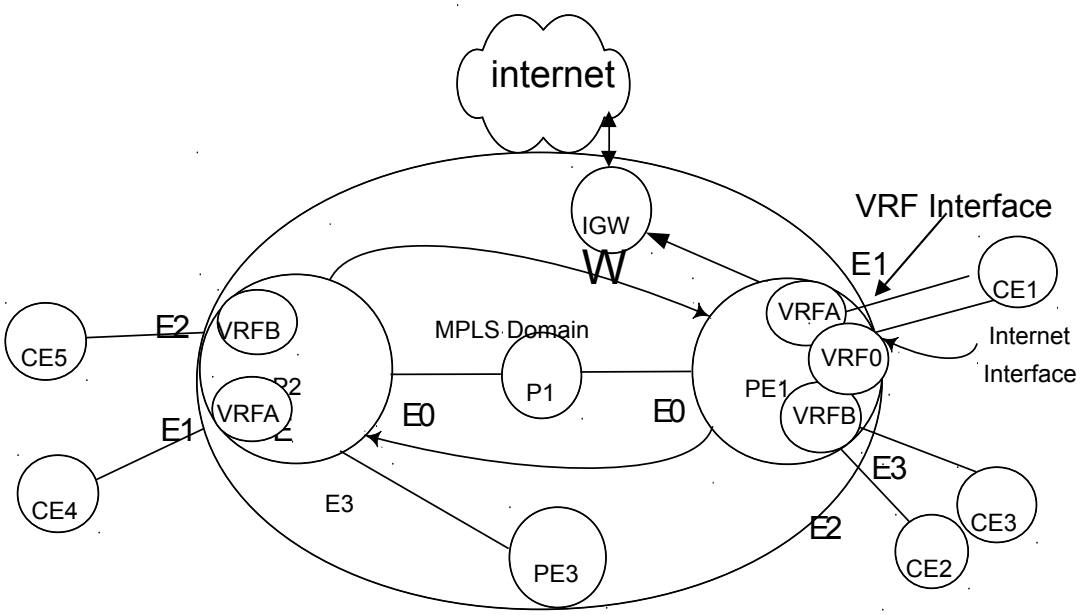

Fig 3-1 Non-VRF Internet Access Mode

4.1.2 VRF Internet Access Mode 1

As demonstrated in the next figure, in VRF Internet Access Mode 1, PE routers communicate with Internet gateways via non-VFP interface. The Internet access traffic of VPN sites and the traffic between VPN sites are sent and received via the private netwrok connections between CE and PE. PE routes contain the whole or part of Internet routes, or only a default routes pointing to the Internet gateway. When the IP packets accessing Internet from VPN reach the VRF interfaces of PE, a failed lookup in the VRF route table will cause a lookup in the global route table. If a match is found, the pakce will be forwarded to the Internet gateway, and finaly to Internet via the gateway. To emable the Internet hosts access VPN sites, a special static route needs to be registered in the PE global route table, whose destiantion segment is the IP address sub-net address which is globally registered in the VPN site, egress interface is the private network interface pointing to the VPN site, and next-hop is CE router. This static route is advertised to the internet gateway by PE, and then to Internet by the gateway. When the packets from the Internet to the VPN reach the pbulic network interface of PE, it will be forwarded to the next-hop via the private network interface if it matches the static route in the PE's global route tabel poinging to the VPN site. In this mode, the global route table and VRF route tabel of PE routers are not completely isolated, since the global one contains part of VPN routes.

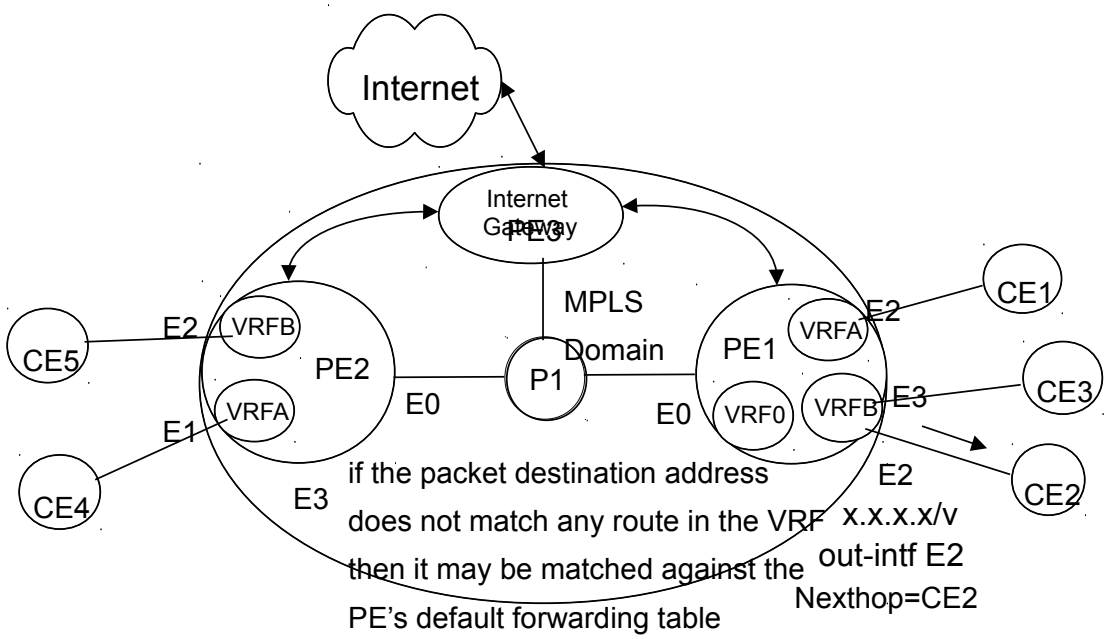

Fig 3-2 VRF Internet Access Mode 1

4.1.3 VRF Internet Access Mode 3

In VRF Internet Access Mode 3, as demonstrated in the next figure, VPN site access the Internet via private network connections between PE and CE. The VRF route tabl eof PE routers contain Internet routes, which are learnt via the PE routers conencted with the Internet gateway (Internet PE). Internet PE will create an Internet VRF, and connect withe Internet gateway withe the interface bound with the Internet VRF. Thus , the Internet gateway will be able to advertise Inernet routes to the VRF route table of the Internet PE. There routes then will be advertised to VRF of other PE routers as VPNv4 routes. PE routers connected with the VPN sites needing to access the internet will advertise correspondng VRF routes (only those routes whose destiatnion segments are globally registered IP addess sub-net in the VPN) to Internet PE via VPNv4 route. These routes will be added into the Internet VRF and then advertised to Internet by the Internet gateway. The import and export policy of these routes depend on the route-target configuration of MBGP and vrf. Please notice that, in this mode, no overlap of address or route is allowed between the VPN sites capable of accessing the Internet.

match any route in the VRF

It is not recommended for users to access the public network in this mode, for a large number of Internet routes will be imported to PE.

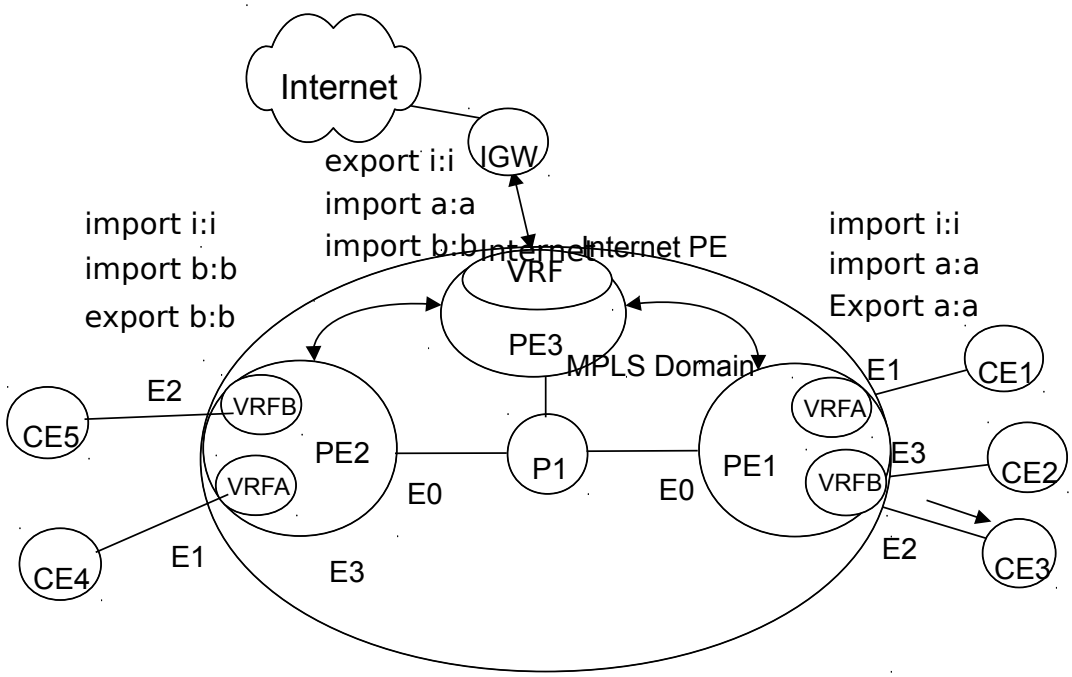

Fig 3-3 VRF Internet Access Mode 3

4.2 Public Network Access Configuration

Public Network Access Configuration Task Sequence:

1. Configure non-VRP Internet access mode
   (1) Configure regular L3VPN
   (2) Add a public connection between CE and PE, the connection interface is a non-VRF one.
   (3) Filter routes on CE; advertise public network routes to PE via the public network connection.
   (4) Configure proper filter policy on the public network interface, to filter the packets whose source and destination addresses are private network addresses.
   (5) Configure default routes
      1) IGW import the default routes to BGP
      2) PE advertise the default routes to CE via the public network connection
      3) CE advertise the default routes to PE via the private network connection, and then to other CE.
   (6) Configure the static route
      1) Configure the static route pointing to Internet on CE1
      2) Configure the static route pointing to the public network interface of CE on PE1
2. Configure VRP Internet access mode 1
   (1) Configure regular L3VPN
   (2) Configure ip vrf forwarding VPNA fallback global on the private network

interface of PE

(3) Configure 3 static routes:

1) Configure a default route on CE, whose next-hop is the proxy server
2) Add a default route to Internet on PE, whose next-hop is IGW. PE advertises a default route via OSPF, whose next-hop is the PE itself.
3) Add a static route form Internet to proxy server to the global route table of PE, whose destination is VPN public network address, next-hop is proxy server; and advertise this route to other PE via OSPF

Configure non-VRF Internet Access Mode

This configuration concerns no extra command line other than the configuration sequence. Please refer to the configuration instruction of the corresponding function for details about commands

Configure VRP Internet access mode 1

1. Configure VRP Internet access mode 1
   (1) Configure regular L3VPN
   (2) Configure ip vrf forwarding VPNA fallback global on the private network interface of PE
   (3) Configure 3 static routes
      1) Configure a default route on CE, whose next-hop is the proxy server
      2) Add a default route to Internet on PE, whose next-hop is IGW. PE advertises a default route via OSPF, whose next-hop is the PE itself.
      3) Add a static route form Internet to proxy server to the global route table of PE, whose destination is VPN public network address, next-hop is proxy server; and advertise this route to other PE via OSPF

Command	Explanation
Configure regular L3VPN	Refer to the BGP MPLS VPN configuration
Interface Configuration Mode	
[no] ip vrf forwarding <*vrf_name*> fallback global	Necessary Configure the global second lookup function of VRF route table. It is not configured by default. Before this configuration, cancel the VRF configuration in the interface view.
Global Configuration Mode	
[no] ip route vrf <*vrf-name*> {<ip-prefix> <mask>|<ip-prefix/<prefix-length>} {<gateway-address>|null0}	Necessary Configure static routes, Only three are needed: one is the default route of CE1, another is the default route to Internet on PE3 and the other is the static route form Internet to the proxy server on PE1.

4.3 Public Network Access Typical Instances

4.3.1 Non-VRF Internet Access Mode

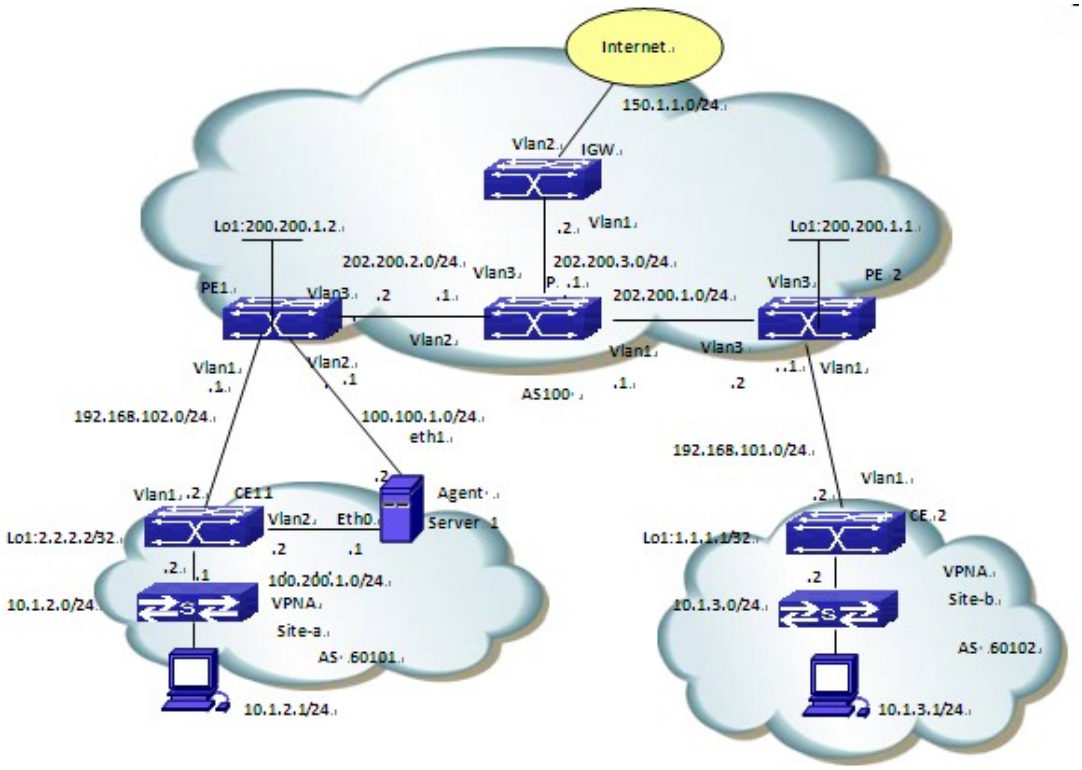

Fig 3-4 Non-VRF Internet Access Mode

The configuration of CE1 is as follows:

```
CE1#config
CE1(config)#access-list 1 deny 100.100.1.0 0.0.0.255
CE1(config)#access-list 1 deny 100.200.1.0 0.0.0.255
CE1(config)#access-list 1 permit any-source
CE1(config)#access-list 2 permit 10.1.1.0 0.0.0.255
CE1(config)#access-list 2 permit 10.1.2.0 0.0.0.255
CE1(config)#access-list 2 deny any-source
CE1(config)# interface vlan 1
CE1(config-if-Vlan1)#ip address 192.168.102.2 255.255.255.0
CE1(config-if-Vlan1)#exit
CE1(config)# interface vlan 2
CE1(config-if-Vlan1)#ip address 100.200.1.2 255.255.255.0
CE1(config-if-Vlan1)#exit
CE1(config)# interface vlan 3
CE1(config-if-Vlan1)#ip address 10.1.2.2 255.255.255.0
CE1(config-if-Vlan1)#exit
```

```
CE1(config)# interface loopback 1
CE1(config-if-Vlan1)#ip address 2.2.2.2 255.255.255.255
CE1(config-if-Vlan1)# exit
CE1(config)#router bgp 60102
CE1(config-router)#network 120.1.1.0/24
CE1(config-router)#network 120.1.2.0/24
CE1(config-router)#network 10.1.2.0/24
CE1(config-router)#redistribute connected
CE1(config-router)#neighbor 100.100.1.1 remote-as 100
CE1(config-router)#neighbor 100.100.1.1 distribute-list 2 out
CE1(config-router)#neighbor 192.168.102.1 remote-as 100
CE1(config-router)#neighbor 192.168.102.1 default-originate
CE1(config-router)#neighbor 192.168.102.1 distribute-list 1 out
CE1(config-router)#exit
CE1(config)# ip route 100.100.1.1 255.255.255.0 100.200.1.1
CE1(config)# ip route 0.0.0.0/0 100.200.1.1
CE1(config)# exit
```

The configuration of PE1 is as follows:

```
PE1#config
PE1(config)#access-list 100 deny ip 10.1.2.0 0.0.0.255 any-destination
PE1(config)#access-list 100 deny ip 10.1.2.0 0.0.0.255 any-destination
PE1(config)#access-list 100 deny ip 10.1.3.0 0.0.0.255 any-destination
PE1(config)#access-list 100 deny ip anysource 200.200.1.0 0.0.0.255
PE1(config)#access-list 100 deny ip anysource 202.200.0.0 0.0.255.255
PE1(config)#firewall enable
PE1(config-vrf)#ip vrf VRF-A
PE1(config-vrf)#rd 100:10
PE1(config-vrf)#route-target both 100:10
PE1(config-vrf)#exit
PE1(config)#interface vlan1
PE1(config-if-Vlan1)#ip vrf forwarding VRF-A
PE1(config-if-Vlan1)#ip address 192.168.102.1 255.255.255.0
PE1(config-if-Vlan1)#exit
PE1(config)#interface vlan2
PE1(config-if-Vlan2)#ip address 100.100.1.1 255.255.255.0
PE1(config-if-Vlan2)#ip access-group 1 in
PE1(config-if-Vlan2)#exit
PE1(config)# interface vlan3
PE1(config-if-Vlan3)#label-switching
PE1(config-if-Vlan3)#ldp enable
PE1(config-if-Vlan3)#ip address 202.200.2.2 255.255.255.0
PE1(config-if-Vlan3)#exit
PE1(config)#interface Loopback1
PE1(config)#ip address 200.200.1.2 255.255.255.255
```

```
PE1(config)#router ospf
PE1(config-router)#network 200.200.1.2/32 area 0
PE1(config-router)#network 202.200.2.0/24 area 0
PE1(config-router)#exit
PE1(config)#router bgp 100
PE1(config-router)#neighbor 100.200.1.2 remote-as 60102
PE1(config-router)#neighbor 200.200.1.1 remote-as 100
PE1(config-router)#neighbor 202.200.3.2 remote-as 100
PE1(config-router)#neighbor 202.200.3.2 next-hop-self
PE1(config-router)#address-family vpnv4 unicast
PE1(config-router-af)#neighbor 200.200.1.1 activate
PE1(config-router-af)#exit-address-family
PE1(config-router)#address-family ipv4 vrf VRF-A
PE1(config-router-af)#neighbor 192.168.102.2 remote-as 60102
PE1(config-router-af)#no neighbor 192.168.102.2 send-community extended
PE1(config-router-af)#exit-address-family
PE1(config-router)#exit
PE1(config)# router ldp
PE1(config-router)#ip route 100.200.1.2 255.255.255.0 100.100.1.2
```

The configuration of P is as follows:

```
P#config
P(config)#interface Vlan1
P(config-if-Vlan1)#label-switching
P(config-if-Vlan1)#ldp enable
P(config-if-Vlan1)#ip address 202.200.1.1 255.255.255.0
P(config-if-Vlan1)#exit
P(config)#interface Vlan2
P(config-if-Vlan2)#label-switching
P(config-if-Vlan2)#ldp enable
P(config-if-Vlan2)#ip address 202.200.2.1 255.255.255.0
P(config-if-Vlan2)#exit
P(config)#interface Vlan3
P(config-if-Vlan3)#ip address 202.200.3.1 255.255.255.0
P(config-if-Vlan3)#exit
P(config)#router ospf
P(config-router)#network 202.200.1.0/24 area 0
P(config-router)#network 202.200.2.0/24 area 0
P(config-router)#network 202.200.3.0/24 area 0
P(config-router)#exit
P(config)#router ldp
```

The configuration of PE2 is as follows:

```
PE2#config
PE2(config)#ip vrf VRF-A
PE2(config-vrf)#rd 100:10
```

```
PE2(config-vrf)#route-target both 100:10
PE2(config-vrf)#exit
PE2(config)#interface Vlan1
PE2(config-if-Vlan1)#ip vrf forwarding VRF-A
PE2(config-if-Vlan1)#ip address 192.168.101.1 255.255.255.0
PE2(config-if-Vlan1)#exit
PE2(config)#interface Vlan2
PE2(config-if-Vlan2)#label-switching
PE2(config-if-Vlan2)#ldp enable
PE2(config-if-Vlan2)#ip address 202.200.1.2 255.255.255.0
PE2(config-if-Vlan2)#exit
PE2(config)#interface Loopback1
PE2(config-if-loopback1)#ip address 200.200.1.1 255.255.255.255
PE2(config-if-loopback1)#exit
PE2(config)#router ospf
PE2(config-router)#network 200.200.1.1/32 area 0
PE2(config-router)#network 202.200.1.0/24 area 0
PE2(config-router)#exit
PE2(config)#router bgp 100
PE2(config-router)#address-family vpnv4 unicast
PE2(config-router-af)#neighbor 200.200.1.1 activate
PE2(config-router-af)#exit-address-family
PE2(config-router)#address-family ipv4 vrf VRF-A
PE2(config-router-af)#neighbor 192.168.101.2 remote-as 60101
PE2(config-router-af)#no neighbor 192.168.101.2 send-community extended
PE2(config-router-af)#exit-address-family
PE2(config-router)#exit
PE2(config)#router ldp
```

The configuration of CE2 is as follows:

```
CE2#config
CE2(config)#interface vlan 1
CE2(config-if-Vlan1)#ip address 192.168.101.2 255.255.255.0
CE2(config-if-Vlan1)#exit
CE2(config)#interface Loopback1
CE2(config-if-Loopback1)#ip address 1.1.1.1 255.255.255.255
CE2(config-if-Loopback1)#exit
CE2(config)#router bgp 60101
CE2(config-router)#network 10.1.3.0/24
CE2(config-router)#neighbor 192.168.101.1 remote-as 100
```

The configuration of IGW is as follows:

```
IGW#config
IGW(config)#interface Vlan1
IGW(config-if-Vlan1)#ip address 202.200.3.2 255.255.255.0
IGW(config-if-Vlan1)#exit
```

IGW(config)#interface Vlan2
IGW(config-if-Vlan2#ip address 150.1.1.1 255.255.255.0
IGW(config-if-Vlan2#exit
IGW(config)#router ospf
IGW(config-router)#network 202.200.3.0 0.0.0.255 area 0
IGW(config-router)#exit
IGW(config)#router bgp 100
IGW(config-router)#neighbor 202.200.2.2 remote-as 100
IGW(config-router)#neighbor 202.200.2.2 default-originate

4.3.2 VRF Internet Access Mode 1

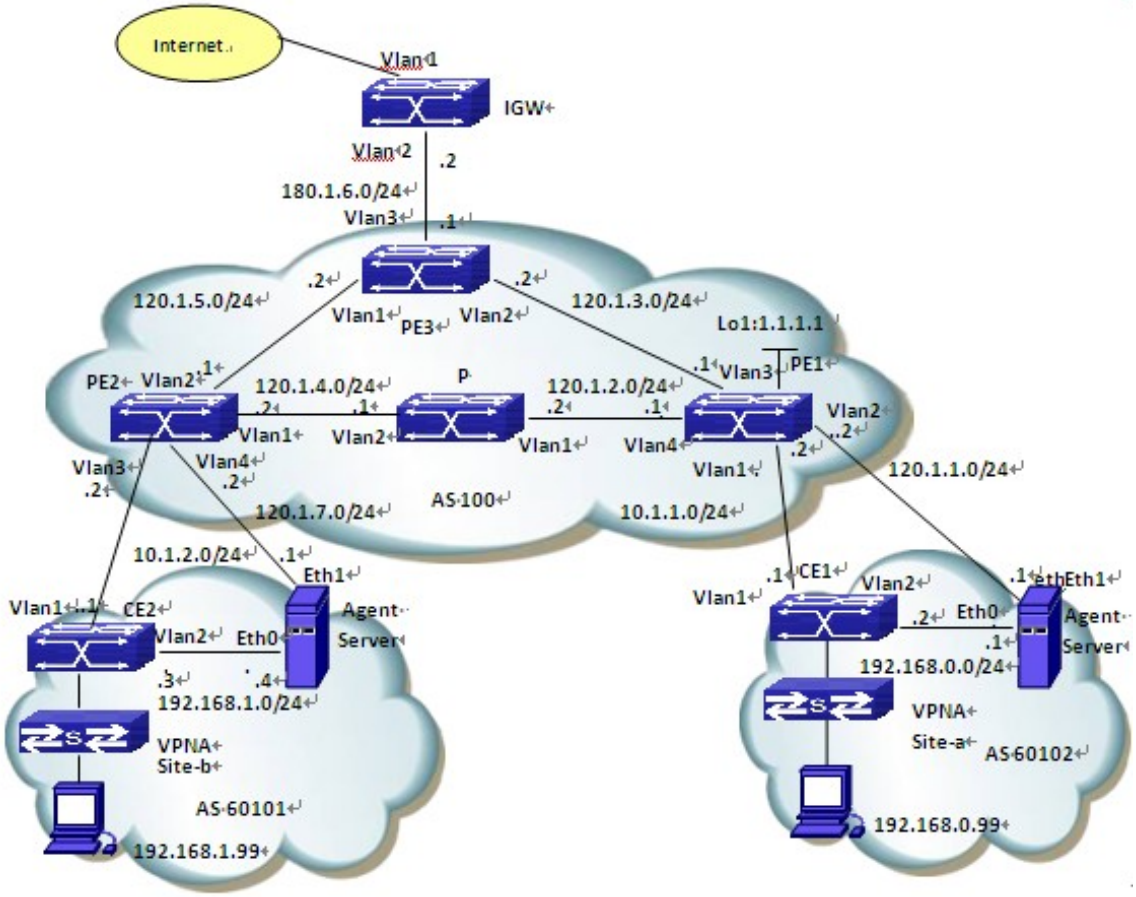

Fig 3-5 VRF Internet Access Mode 1

Site-a and site-b belong to VPNA; their users can intercommunicate and all need to access the Internet. Configure proxy servers separately in site-a and site-b to realize NAT when their users access Internet with the private network addresses.

The configuration of CE1 is as follows:

CE1#config
CE1(config)#interface Vlan1
CE1(config-if-Vlan1)#ip address 10.1.1.1 255.255.255.0

```
CE1(config-if-Vlan1)#exit
CE1(config)#interface Vlan2
CE1(config-if-Vlan2)#ip address 192.168.0.2 255.255.255.0
CE1(config-if-Vlan2)#exit
CE1(config)#interface loopback1
CE1(config-if-Loopback1)#ip address 11.11.11.11 255.255.255.255
CE1(config-if-Loopback1)#exit
CE1(config)#ip route 0.0.0.0/0 192.168.0.1
CE1(config)#router bgp 60101
CE1(config-router)#neighbor 10.1.1.2 remote-as 100
CE1(config-router)#network 192.168.0.0/24
```

The configuration of PE1 is as follows:

```
PE1#config
PE1(config)#ip vrf VPNA
PE1(config-vrf)#rd 100:10
PE1(config-vrf)#route-target both 100:10
PE1(config-vrf)#exit
PE1(config)#interface Vlan1
PE1(config-if-Vlan1)#ip vrf forwarding VPNA
PE1(config-if-Vlan1)#ip address 10.1.1.1 255.255.255.0
PE1(config-if-Vlan1)#exit
PE1(config)#interface Vlan2
PE1(config-if-Vlan2)#ip vrf forwarding VPNA fallback global
PE1(config-if-Vlan2)#ip address 120.1.1.2 255.255.255.0
PE1(config-if-Vlan2)#exit
PE1(config)#interface Vlan3
PE1(config-if-Vlan3)#ip address 120.1.3.1 255.255.255.0
PE1(config-if-Vlan2)#exit
PE1(config)#interface Vlan4
PE1(config-if-Vlan4)#label-switching
PE1(config-if-Vlan4)#ldp enable
PE1(config-if-Vlan4)#ip address 202.200.1.2 255.255.255.0
PE1(config-if-Vlan4)#exit
PE1(config)#interface loopback1
PE1(config-if-Loopback1)#ip address 1.1.1.1 255.255.255.255
PE1(config-if-Loopback1)#exit
PE1(config)#router ospf
PE1(config-router)#redistribute static
PE1(config-router)#network 1.1.1.1/32 area 0
PE1(config-router)#network 120.1.2.0/24 area 0
PE1(config-router)#network 120.1.3.0/24 area 0
PE1(config-router)#exit
PE1(config)#router bgp 100
PE1(config-router)#neighbor 2.2.2.2 remote-as 100
```

```
PE1(config-router)#neighbor 2.2.2.2 update-source 1.1.1.1
PE1(config-router)#address-family vpnv4 unicast
PE1(config-router-af)#neighbor 2.2.2.2 activate
PE1(config-router-af)#exit-address-family
PE1(config-router)#address-family ipv4 vrf VPNA
PE1(config-router-af)#network 120.1.1.0/24
PE1(config-router-af)#neighbor 10.1.1.1 remote-as 60101
PE1(config-router-af)#no neighbor 10.1.1.1 send-community extended
PE1(config-router-af)#exit-address-family
PE1(config-router)#exit
PE1(config)#router ldp
PE1(config-router)#exit
PE1(config)#ip route 120.1.1.0/24 vlan 2 120.1.1.1
```

The configuration of P is as follows:

```
P#config
P(config)#interface Vlan1
P(config-if-Vlan1)#label-switching
P(config-if-Vlan1)#ldp enable
P(config-if-Vlan1)#ip address 120.1.2.2 255.255.255.0
P(config-if-Vlan1)#exit
P(config)#interface Vlan2
P(config-if-Vlan2)#label-switching
P(config-if-Vlan2)#ldp enable
P(config-if-Vlan2)#ip address 120.1.4.1 255.255.255.0
P(config-if-Vlan2)#exit
P(config)#router ospf
P(config-router)#network 0.0.0.0/0 area 0
P(config-router)#exit
P(config)#router ldp
```

The configuration of PE2 is as follows:

```
PE2#config
PE2(config)#ip vrf VPNA
PE2(config-vrf)#rd 100:10
PE2(config-vrf)#route-target both 100:10
PE2(config-vrf)#exit
PE2(config)#interface Vlan1
PE2(config-if-Vlan1)#label-switching
PE2(config-if-Vlan1)#ldp enable
PE2(config-if-Vlan1)#ip address 120.1.4.2 255.255.255.0
PE2(config-if-Vlan1)#exit
PE2(config)#interface Vlan2
PE2(config-if-Vlan2)#ip address 120.1.5.1 255.255.255.0
PE2(config-if-Vlan2)#exit
PE2(config)#interface Vlan3
```

```
PE2(config-if-Vlan3)#ip vrf forwarding VPNA
PE2(config-if-Vlan1)#ip address 10.1.2.2 255.255.255.0
PE2(config-if-Vlan1)#exit
PE2(config)#interface Vlan4
PE2(config-if-Vlan4)#ip vrf forwarding VPNA fallback global
PE2(config-if-Vlan4)#ip address 120.1.7.2 255.255.255.0
PE2(config-if-Vlan4)#exit
PE2(config)#interface Loopback1
PE2(config-if-Loopback1)#ip address 2.2.2.2 255.255.255.255
PE2(config-if-Loopback)#exit
PE2(config)#router ospf
PE2(config-router)#redistribute static
PE2(config-router)#network 2.2.2.2/32 area 0
PE2(config-router)#network 120.1.4.0/24 area 0
PE2(config-router)#network 120.1.5.0/24 area 0
PE2(config-router)#exit
PE2(config)#router bgp 100
PE2(config-router)#neighbor 1.1.1.1 remote-as 100
PE2(config-router)#neighbor 1.1.1.1 update-source 2.2.2.2
PE2(config-router)#address-family vpnv4 unicast
PE2(config-router-af)#neighbor 1.1.1.1 activate
PE2(config-router-af)#exit-address-family
PE2(config-router)#address-family ipv4 vrf VPNA
PE2(config-router-af)#network 120.1.7.0/24
PE2(config-router-af)#neighbor 10.1.2.1 remote-as 60102
PE2(config-router-af)#no neighbor 10.2.1.1 send-community extended
PE2(config-router-af)#exit-address-family
PE2(config-router)#exit
PE2(config)#router ldp
PE2(config-router)#exit
PE2(config)#ip route 120.1.7.0/24 vlan 4 120.1.7.1
```

The configuration of PE3 is as follows:

```
PE3#config
PE3(config)#interface Loopback1
PE3(config-if-Loopback1)#ip address 3.3.3.3 255.255.255.255
PE3(config-if-Loopback1)#exit
PE3(config-if-Vlan1)#interface Vlan1
PE3(config-if-Vlan1)#ip address 120.1.5.2 255.255.255.0
PE3(config-if-Vlan1)#exit
PE3(config)#interface Vlan2
PE3(config-if-Vlan2)#ip address 120.1.3.2 255.255.255.0
PE3(config-if-Vlan2)#exit
PE3(config)#interface Vlan3
PE3(config-if-Vlan3)#ip address 180.1.6.1 255.255.255.0
```

```
PE3(config-if-Vlan3)#exit
PE3(config)#router ospf 1
PE3(config-router)#default-information originate
PE3(config-router)# network 0.0.0.0/0 area 0
PE3(config-router)#exit
PE3(config)#router bgp 100
PE3(config-router)#network 120.1.1.0 mask 255.255.255.0
PE3(config-router)#network 120.1.7.0 mask 255.255.255.0
PE3(config-router)#neighbor 180.1.6.2 remote-as 200
PE3(config-router)#exit
PE3(config)#ip route 0.0.0.0/0 180.1.6.2
```

The configuration of CE2 is as follows:

```
CE2#config
CE2(config)#interface Vlan1
CE2(config-if-Vlan1)#ip address 10.1.2.1 255.255.255.0
CE2(config-if-Vlan1)#exit
CE2(config)#interface Vlan2
CE2(config-if-Vlan2)#ip address 192.168.1.3 255.255.255.0
CE2(config-if-Vlan2)#exit
CE2(config-if-Loopback1)#interface Loopback1
CE2(config-if- Loopback1)#ip address 22.22.22.22 255.255.255.255
CE2(config-if- Loopback1)#exit
CE2(config)#ip route 0.0.0.0/0 192.168.1.4
CE2(config)#router bgp 60101
CE2(config-router)#neighbor 10.1.2.2 remote-as 100
CE2(config-router)#network 192.168.1.0/24
CE2(config-router)#exit
```

The configuration of IGW is as follows:

```
IGW#config
IGW(config)#interface Vlan1
IGW(config-if-Vlan1)#ip address 180.1.5.2 255.255.255.0
IGW(config-if-Vlan1)#exit
IGW(config)#interface Vlan2
IGW(config-if-Vlan2)#ip address 180.1.6.2 255.255.255.0
IGW(config-if-Vlan2)#exit
IGW(config)#router bgp 200
IGW(config-router)#neighbor 180.1.6.1 remote-as 100
IGW(config-router)#exit
```

4.4 Public Network Access Troubleshooting

When configuring and using Public Network Access, some problems like incorrect physical connections, configuration errors may cause it to fail, so please pay attention to

the following notices to avoid them:

- ☞ First, make sure the regular MPLS BGP VPN works correctly, and the intercommunication is normal in the private network. If the communication in VPN fails, please refer to the help on MPLS BGP VPN troubleshooting.
- ☞ Second, check the public network access mode in use is non-VRF or VRF, for their configurations differ a lot.
- ☞ In non-VRF mode, please remember to configure filter policy on the non-VRF interface of PE-CE, to block the private network route and traffic from entering PE through the public network interface. Otherwise, there might be security threats. Besides, make sure the advertisement of default routes and the NAT configuration to IGW are correct.
- ☞ In VRF mode, please make sure to use “ip vrf forwarding vrf_name fallback global” command while configuring the private network interface, to prevent look up the global route table for a second time if the attempt to find the private network route fails. Besides, make sure the advertisement of default routes and the NAT configuration to IGW are correct.
- ☞ At last, if all above steps are correct, CE will be able to access Internet. No matter which networking mode mentioned above is used, other CE access Internet after forwarding traffic to PE via VPN; the traffic from Internet should also be forwarded after passing PE.

Chapter 5 VPLS

5.1 VPLS Overview

Nowadays IP network has been all over the world. How to use the existing IP network to offer a low-cost private network that becomes a focuss to providers. As a result, MPLS VPN—a technology which can offer VPN service in IP network, easy to set rate and configure simply. There are two types in this technology: MPLS L3VPN and MPLS L2VPN. Since MPLS L3VPN intervenes users' internal route management, provider's management is complex. MPLS L2VPN in a traditional way of VLL, offer a point-to-point L2VPN service, it can make the connection between two stations liking the direct link connection. However, it cannot offer the switchover between multi-points for providers. VPLS develops on the basis of the traditional MPLS L2VPN, it can implement VPN network (multi-points to multi-points). So VPLS provides a better solution for providers.

VPLS, namely Virtual Private LAN Service, is a service to offer LAN in MPLS network, accurately, it's a L2VPN technology based on Ethernet. VPLS technology includes the advantages of Ethernet and MPLS, and it can make user's network communicate with each other at different location, as they are connected with each other directly. Furthermore, VPLS enables users to extend their LAN to MAN, or even WAN.

VPWS is a special example of VPLS, conversely, VPLS is extension of VPWS. Since VPLS offers a VPN group network (multi-points to multi-points) while VPWS only offers a point-to-point L2VPN solution, in this way ,VPLS networks can be looked as a switch to user CE, but VPWS correspond as an Ethernet cable.

VPLS chooses LDP signaling and BGP signaling to form PW. A signaling based on LPD protocol establishes virtual circuit by establishing a point-to-point LDP session between two PEs, since LDP protocol is set earlier and application is simple, many products support this protocol. In terms of compatibility, we adopt a signaling mechanism based on LDP.

5.1.1 Basic Concept of VPLS

- **VPLS（Virtual Private LAN Service）**

VPLS is a service to offer LAN in MPLS network, it can make users to access network from dispersed points at the same time, as they are connected with each other directly. Furthermore, VPLS enables users to extend their LAN to MAN, or even WAN.

- **VC（Virtual Circuit）**

It is a unilateralism logic connection between two points, a pair of reverse VC forms a PW.

- **S-TAG(Service Tag)、S-VID(Service VLAN ID)**

They are offered by provider's network to identify users.

- **P-TAG(Provider Tag)、P-VID(Provider VLAN ID)**

  They are expected VLAN Tags for the other peer.

- **VFI（Virtual Forward Instance）**

  Each VFI offers separated VPLS service, and it fulfills the function of Ethernet bridging. Using VFI can map VPLS's actual access links to each PW, in turn, it can push PW to the end and map PW to the actual access link.

- **UPE（User facing-Provider Edge）**

  It is an aggregation device for accessing VPN.

- **NPE（Network Provider Edge）**

  It is a core PE device at the core field edge of VPLS network, offer the transparent VPLS transmission between core networks.

- **MTU（Maximum Transmission Unit）**

  Maximum transmission unit.

- **QinQ（802.1Q in 802.1Q）**

  It is a tunnel protocol based on 802.1Q encapsulation, it is able to offer point to multi-point service for L2VPN. It encapsulates user's VLAN Tag in private network to public network, then the final packets with two tags go through provider's backbone network, which provides users with a simpler layer 2 VPN tunnel.

- **PW Signaling**

  It is used to establish and maintain PW, and is the basis of VPLS. It can be used to discover VFI's peer PE device automatically. There are LDP and BGP in PW signaling so far. This work supports LDP only.

- **VPWS (Virtual Private Wire Service)**

  It is a point-to-point service for L2VPN. When one peer (AC or PW) receives packets, it will be forwarded directly to the other peer (PW or AC). Compared with VPLS, VPWS does not require to learn and look for MAC address, furthermore, there is no broadcast, multicast, etc., so it is more efficient.

5.1.2 Basic Network Model of VPLS

The network model of VPLS has five parts: CE、PE、P、AC and PW. The following is the basic concept.

- **CE（Custom Edge）**

  It is an edge device connected directly with providers. It can be a router, a switch or a host.

- **PE（Provider Edge）**

  It is an edge device offered by providers, connected with CE, and responsible for accessing VPN. It can map and forward packets from public tunnel to private tunnel.

- **P（Provider）**

  It a backbone router in provider's network, it does not connect with CE. It only needs to have basic MPLS forwarding capability and does not maintain VPN information.

- **AC（Attachment Circuit）**

In L2VPN, CE is able to access to PE via AC. AC is used to transport frames between CE and PE, and it can be a physic or logical link.

- **PW（Pseudo Wire）**

Simply, PW means VC and tunnel, which can be LSP, GRE or CR-LSP. For VPLS, PW is like a direct tunnel between ACs to complete layer 2 data passthrough of users.

There are two models in rfc4762: one is fully connection to PW and the other is layered VPLS model. As shown in figure 1 and 2

5.1.3 Advantages of VPLS

VPLS advantages:

- VPLS uses the Ethernet interface to users, simplifying LAN/WAN boundary, can support a fast and flexible service deployment.
- VPLS gives the right of controlling and maintaining route policy to users, simplifying the network management from providers.
- All CEs included in VPLS service are part of a sub-network, simplifying IP address configuration.
- VPLS service does not participate in IP addressing and route.

5.1.4 Basic VPLS Network Model of Fully PE Connection

All PEs connect with each other logically, they can learn MAC addresses and forward packets among multi-points. MPLS network offers tunnel to pass through packets between VPN stations. P devices are similar to those in L3VPN, they are responsible for forwarding MPLS packets but do not participate in learning or exchanging MAC addresses. In order to overlap MAC addresses in VPNs, the forwarding tables are independent.

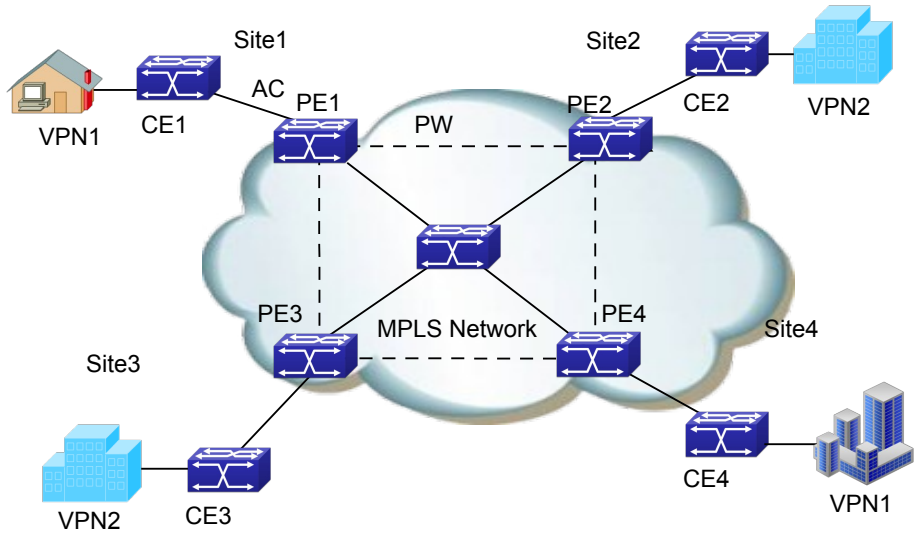

Fig 4-1 VPLS model of fully connection PE

Fully connection VPLS model avoids loopback through level division forwarding. If a PE receives packet from a PW, then this packet will not be forwarded to other PWs relevant to VFI. That is to say, any two PEs communicate with each other via a PW directly, do not forward packet via the third PE. That is why VFIs require fully connection PW.

5.1.5 Layered VPLS Model

All NPEs connect with each other logically. UPE establishes a virtual connection with closest NPE, and exchange packets with VPN station through NPE, so network topology is getting stratificational. UPE is mostly used to access VPN and has lower performance requirement, while NPE is mostly used to aggregate flow, and has higher performance requirement. Additionally, to ensure a strong network, we can add a link backup between UPE and NPE. The virtual connection between UPE and NPE can be established according to QinQ or LDP.

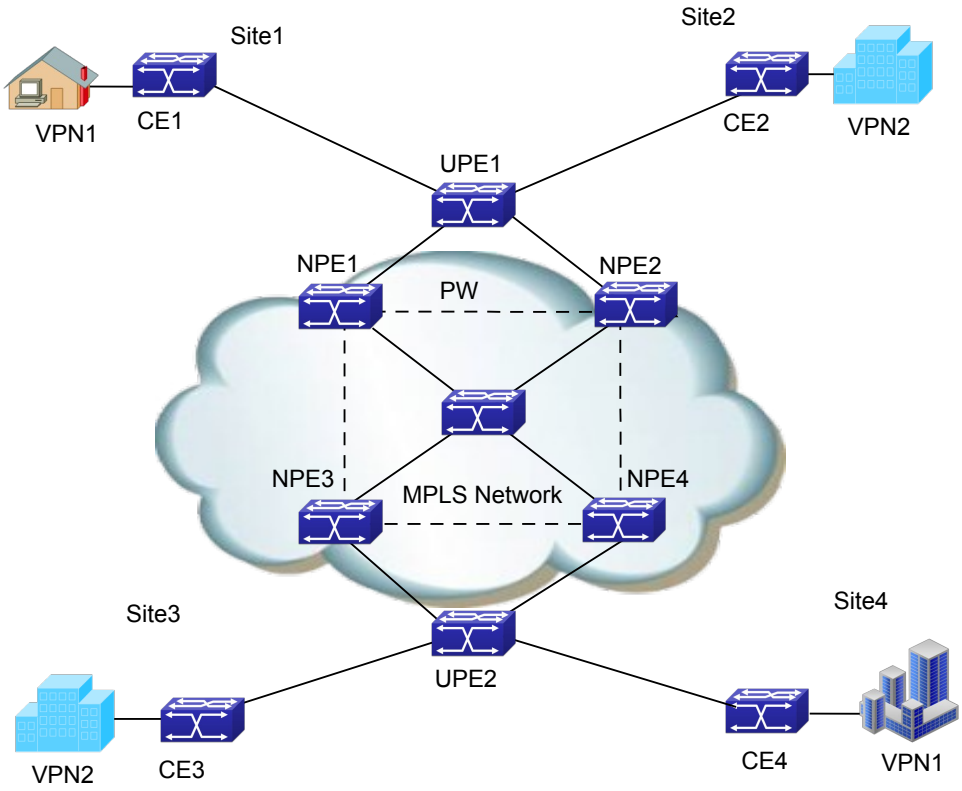

Fig 4-2 Layered VPLS model

According to the different connections between UPE and NPE, H-VPLS is divided to LSP and QinQ access method.

- **LSP access method**

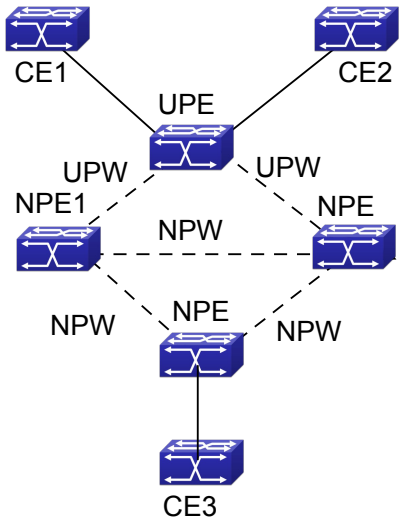

Fig 4-3 LSP access method

As shown in figure, UPE works as an aggregation device, it connects NPE1 and

NPE2 via LSP. UPE must establish virtual U-PW with NPE1 and NPE2 separately (U-PW connection needs the new VFI on UPE and NPE to set peers, and PWIDs on two devices must be the same), UPE does not establish the virtual connection with other devices.

The process of forwarding data for LSP access mode is as follows:

(1) UPE sends packet coming from CE to NPE1, and marks the corresponding VC tag (it is assigned by NPE1, and work as an unattached tag for multiplex PW) related to U-PW.

(2) When NPE1 receives packet, it will analyze VC tag and decide which VFI it belongs to, then push it in the corresponding VC tag according to destination MAC, finally, forward packet.

(3) After NPE1 receives packet from N-PW, it marks the corresponding VC tag, then forward packet to UPE, finally to CE.

When data exchanging between CE1 and CE2 becomes the exchanging between the local CEs, since UPE has a bridging function, it can complete packet forwarding between CEs. However, for unknown data packets or broadcast packets of destination MAC, UPE still forward them to NPE1 via U-PW when packets are broadcasted to CE2 via bridge. Finally NPE1 will copy and forward packets to CEs on the other peer.

- **QinQ access method**

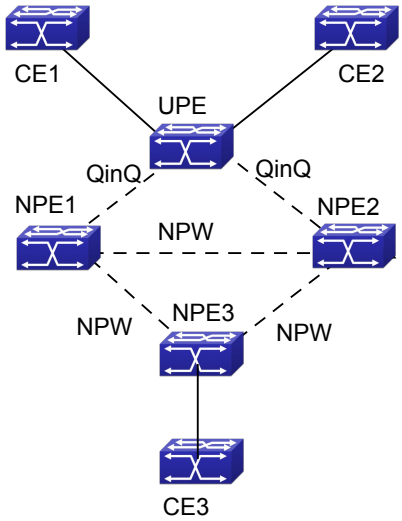

Fig 4-4 QinQ access method

As shown in figure, UPE is a standard bridging device, it establishes point-to-point Ethernet QinQ connection between UPE and PEs (that is, apply QinQ to CE interface while apply VLAN access mode to PE1). Packets received by UPE will be marked an outer VLAN tag, when they are forwarded to PE1, the outer VLAN tag can be described to a VLAN tag of providers according to VLAN access mode, namely, a service delimiter, according to the delimiter, packets are mapped to relevant VFI, then this VFI makes up its mind how to forward packets (unicast or multicast).

The process of forwarding data for QinQ is as follows:

(1) Enable QinQ at CE access port, add the unattached multiplex tag for packets, and pass through them to PE1 via QinQ tunnel between UPE and PE1.

(2) When PE1 receives packet, it will analyze the VLAN tag and decide which VFI it

belongs to, then push relevant PW tag in according to destination MAC address, finally, it will be forwarded.

(3) After PE1 receives packet from PW, it will decide which VFI it belongs to according to PW tag, and marks a VLAN tag according to destination MAC, then forward packet via QinQ tunnel to UPE, finally to CE.

When data exchanging between CE1 and CE2 becomes the exchanging between the local CEs, since UPE has a bridging function, it can complete packet forwarding between CEs. However, for unknown data packets or broadcast packets of destination MAC, UPE still forward them to PE1 via QinQ when packets are broadcasted to CE2 via bridge. Finally PE1 will copy and forward packets to CEs on the other peer.

- **Method of avoidance loopback for H-VPLS model**

Compared to full connection, method of avoidance loopback for the layered VPLS model is a different. Since H-VPLS only needs to establish a full connection between NPEs, do not establish that between UPE and NPE, packets received from PW are not forwarded to relevant PW connected with other NPE, however, it allows to forward packets to relevant PW connected with UPE. Furthermore, packets received from PW connecting to UPE can be forwarded to PW connected with other NPEs.

5.1.6 Packet Forwarding of VPLS

In VPLS model, there are two packets encapsulation methods in AC and PW.

In AC there are VLAN and Ethernet access methods. Introduction is as follows:

- **VLAN access**

Ethernet frame header with a VLAN TAG sent by CE to PE or PE to CE, it is a delimiter added by provider's device to distinguish users, and we call it as S-TAG.

- **Ethernet access**

Ethernet frame header without a delimiter sent by CE to PE or PE to CE. If there is a VLAN TAG in it, however, this TAG is an inner one, and it is meaningless to PE. We call it as U-TAG.

There are two packets encapsulation methods in PW, they are Raw and Tagged modes. Introduction is as follows:

- **Raw mode**

A frame transported by PW cannot without S-TAG: for packets of CE, if packets with delimiter are received, they will remove the delimiters and push PW and tunnel tags in before forwarding. If packets without delimiter are received, they will push PW and tunnel tags in directly before forwarding. For downlink packets of PE, they can add delimiters or not to forward to CE according to the actual configuration, but they cannot rewrite or remove the existent tags.

- **Tagged mode**

A frame transported in PW has to own an S-TAG: for packets of CE, if packets with delimiter are received, it will be compared with the expected VLAN TAG from the peer. If they are same, keep S-TAG, else rewrite S-TAG as the expected VLAN TAG. If the other

peer does not transfer the expected VLAN TAG, it will fill in null TAG (value of TAG is 0). Finally push PW and tunnel tags in before forwarding. On the condition that there is not expected VLAN TAG, TAG is null (TAG equals 0). If packets without delimiter are received, add an expected VLAN TAG or a null TAG, and then push PW and tunnel tags in before forwarding. For downlink packets of PE, they can rewrite, remove or keep delimiters according to the actual configuration before forwarding.

According to the different of AC access mode and packet encapsulation mode in PW, VPLS packet forwarding mode can be divided into the following four types.

- **Forward packets in Ethernet access Raw mode**

As shown in figure, AC uses Ethernet access mode, and packet encapsulation mode is Raw mode on PW, the process of packet forwarding from CE1 to CE4 is described as follows:

1. CE1 sends packets with U-Tag to PE1
2. PE1 chooses an appropriate PW according to destination MAC address and adds a VC tag in packets.
3. In order to forward packets in public network via MPLS tunnel, PE1 adds a public network tunnel tag and forwards packets to PE4.
4. After PE4 receives packets, it finds out the relevant VFI according to the VC tag, and sends packets with U-TAG to CE4.

- **Forward packets in Ethernet access Tagged mode**

As shown in figure, AC uses Ethernet access mode. When packet encapsulation mode is Tagged mode on PW, the process is similar to packets forwarded in Ethernet access mode and Raw mode. The difference is that frame in PW has to be with S-TAG. After PE1 receives packets without S-TAG, first, it will add an expected VLAN TAG or a null TAG, and then, it will push two layers MPLS tag before forwarding. When PE4 receives packets, it will remove the two layers tag and S-TAG before forwarding them to CE4.

- **Forward packets in VLAN access Raw mode**

As shown in figure, AC uses VLAN access mode, messages packet encapsulation mode is RAW mode, the process of packets forwarded from CE1 to CE4 is described as follows:

1. Packets sent by CE1 have to be with S-TAG. If packets forwarded by CE1 without VLAN TAG or with unmatched VLAN TAG, it executes the normal layer 2 forwarding process.
2. After PE1 receives packets, it removes S-TAG and adds a two layer MPLS tags before forwarding them to PE4 via public network MPLS tunnel.
3. After PE4 removes two layer MPLS tags in packets, adds S-TAG before forwarding them to CE4.

- **Forward packets in VLAN access Tagged mode**

As shown in figure, AC uses VLAN access mode. When packet encapsulation mode is Tagged mode on PW, the process is similar to packets forwarded in VLAN access mode and Raw mode. The difference is that frame in PW with S-TAG. After PE1 receives packets, it will be compared with the expected VLAN TAG sent by the peer PE, if they are

same, keep S-TAG, else rewrite S-TAG as the expected VLAN TAG. On the condition that there is not expected VLAN TAG, it will fill in a null TAG (value of TAG is 0).

5.2 VPLS Configuration

VPLS configuration task sequence:
1. Globally enable route protocol (required)
2. Basic configuration of MPLS (required)
   (1) Enable MPLS globally
   (2) Enable tag exchanging on interface
3. Configure LDP session (required)
   (1) Enter ldp view in configuration mode, which stands for enabling LDP function
   (2) Configure remote LDP neighbor and enable targeted-peer command
   (3) Enable LDP command in layer 3 interface view
4. Create PW template (optional)
   (1) Create pw-class and enter pw view
   (2) Set transmission mode for pw-class
5. Configure VFI (required)
   (1) Create VFI, specify VFI ID and enter VFI view
   (2) Configure the end point for remote PW (usually it is same as targeted peer of LDP)
   (3) Specify pw-class for each peer
   (4) Set PW encapsulation mode for VFI
6. Configure users' access mode and bind VFI (VPLS required)
   (1) Enter port view
   (2) Configure binding VFI for port
7. Configure binding PW for port (VPWS required)
   (1) Create L2VC and configure VPWS's peer PE
   (2) Enter port view
   (3) Bind port with the specified PW and enter access mode

1. Globally enable route protocol (required)

Command	Description
Global configuration mode	
router ospf	Enter routing configuration mode for OSPF
OSPF routing configuration mode	
network 0.0.0.0/0 area 0	Configure the interface segment address enabled OSPF, by default, OSPF is disabled on interface.

2. Global MPLS (required)

Command	Description
Global configuration mode	

mpls enable **no mpls enable**	Enable MPLS protocol, no command disables MPLS protocol.
Layer 3 interface view	
label-switching enable **no label-switching**	Enable tags exchanging function, no command disables the function by default.

3. Configure LDP session (required)

(1) Enter ldp view in configuration mode, which stands for enabling LDP function

(2) Configure remote LDP neighbor and enable targeted-peer command

(3) Enable LDP command in layer 3 interface view

Command	Description
Global configuration mode	
router ldp **no router ldp**	Enable LDP protocol, no command disables LDP.
targeted-peer 1.1.1.1	Configure remote LDP neighbor. By default, there is no remote LDP neighbor.
Interface configuration mode	
ldp {enable | disable}	Enable LDP protocol on interface, no command disables LDP protocol.

4. Create PW template (optional)

(1) Create pw-class and enter pw view

(2) Set transmission mode for pw-class

Command	Description
Global configuration mode	
pw-class <pw-class-name> **no pw-class <pw-class-name>**	Create pw-class. By default, there is no PW template.
PW template configuration mode	
transport-mode {ethernet | vlan}	Configure packet encapsulation mode on PW template, ethernet corresponds to raw mode and vlan corresponds to tagged mode.

5. Configure VFI (required)

(1) Create VFI, specify VFI ID and enter VFI view

(2) Configure the end point for remote PW (usually it is same as targeted peer of LDP)

(3) Specify pw-class for each peer

(4) Set PW encapsulation mode for VFI

Command	Description
Global configuration mode	
vfi <vfi-name> <vfi-id> **no vfi <vfi-name>**	Create VFI and specify VFI ID. There is no VFI.
VFI configuration mode	

peer ip-address [pw-id pw-id] [no-split-horizon] [pw-class class-name]	Peer PE included in VPLS instance can configure whether enable level division (enabled by default) and pw template.
transport-mode {ethernet | vlan}	Configure packet encapsulation mode in PW.

6. Configure users' access mode and bind VFI (VPLS required)
   (1) Enter port view
   (2) Configure binding VFI for port

Command	Description
Port configuration mode	
xconnect vfi vfi-id [mode {ethernet | vlan [svid svid]}]	Configure a port to bind VFI and configure AC access mode. By default, port does not bind any VFI.

7. Configure binding PW for port (VPWS required)
   (1) Create L2VC and configure VPWS's peer PE
   (2) Enter port view
   (3) Bind port with the specified PW and enter access mode

Command	Description
Global configuration mode	
l2-vc ip-address pw-id pw-id [group group-id] [pw-class class-name]	Create L2VC and configure VPWS's peer PE. By default, there is no configuration for peer PE.
Enter port view	
xconnect l2-vc pw-id <pw-id> [mode {ethernet | vlan [svid <svid>]}]	Bind port with the specified PW and enter access mode. By default, port is not bound to any PW.

5.3 Typical Examples of VPLS

5.3.1 Full Connection for VPLS Configuration

5.3.1.1 Network Requirement

- CE1 and CE2 belong to two different stations, while they belong to the same VPN1
- CE1 and CE2 can be accessed to PE via port Eth1/0/1
- CE1 can access to PE1 in Ethernet mode, while CE2 can access to PE2 in VLAN mode, Svid equals 200.
- Packet encapsulation mode between PE1 and PE2 is RAW (Ethernet) mode, while that between PE3 and PE1 (PE2) is Tagged (Vlan) mode.

- It requires that establish a layer 2 VPN1 by configuring VPLS, making CE1 and CE2 access each other in layer 2 mode.

5.3.1.2 Group Network Diagram

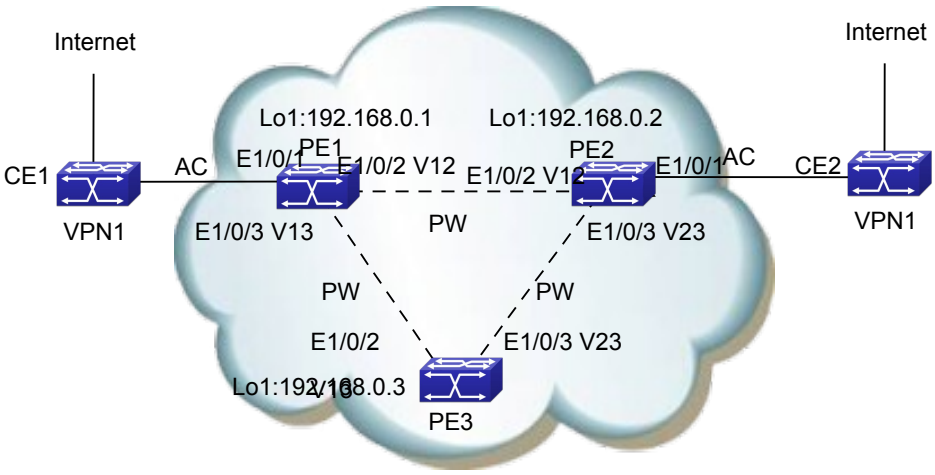

Fig 4-5 full connection VPLS model for PE

5.3.1.3 Configuration Steps

Please configure IP address and mask for interfaces including VLAN and Loopback according to the above figure. The specific configuration is omitted. This is for VPLS examples in LDP.

(1) PE1 configuration

#Create PW template

PE1(config)#pw-class c1

PE1(config-class) #transport-mode ethernet

PE1(config-class)#exit

PE1(config)#pw-class c2

PE1(config-class)#transport-mode vlan

PE1(config-class)#exit

#Configure the routing

PE1(config)#router ospf

PE1(config-router)#router-id 192.168.0.1

PE1(config-router)#network 0.0.0.0/0 area 0

PE1(config-router)#exit

#Configure MPLS capability and label switching capability (Interface connected with CE enables label switching capability in non-H-VPLS)

PE1(config)#mpls enable

```
PE1(config)#int vlan 12
PE1(config-if-vlan12)#label-switching
PE1(config-if-vlan12)#exit
PE1(config)#int vlan 13
PE1(config-if-vlan13)#label-switching
PE1(config-if-vlan13)#exit

#Configure LDP
PE1(config)#router ldp
PE1(config-router)#router-id 192.168.0.1
PE1(config-router)#targeted-peer 192.168.0.2
PE1(config-router)#targeted-peer 192.168.0.3
PE1(config-router)#exit
PE1(config)#int vlan 12
PE1(config-if-vlan12)#ldp enable
PE1(config-if-vlan12)#exit
PE1(config)#int vlan 13
PE1(config-if-vlan13)#ldp enable
PE1(config-if-vlan13)#exit

#Configue instance V1 and remote PE in LDP
PE1(config)#vfi v1 100
PE1(config-vfi)#peer 192.168.0.2 pw-class c1
PE1(config-vfi)#peer 192.168.0.3 pw-class c2
PE1(config-vfi)#exit

#Bind VPLS instance mode as ethernet
PE1(config-if-ethernet1/0/1)#xconnect vfi 100 mode ethernet
```

(2) PE2 configuration

```
#Create PW template
PE2(config)#pw-class c1
PE2(config-class)#transport-mode ethernet
PE2(config-class)#exit
PE2(config)#pw-class c2
PE2(config-class)#transport-mode vlan
PE2(config-class)#exit

#Configure the routing
PE2(config)#router ospf
PE2(config-router)#router-id 192.168.0.2
PE2(config-router)#network 0.0.0.0/0 area 0
PE2(config-router)#exit
```

#Configure MPLS capability and label switching capability (Interface connected with CE does not enable label switching capability in non-H-VPLS)
PE2(config)#mpls enable
PE2(config)#int vlan 12
PE2(config-if-vlan12)#label-switching
PE2(config-if-vlan12)#exit
PE2(config)#int vlan 23
PE2(config-if-vlan23)#label-switching
PE2(config-if-vlan23)#exit

#Configure LDP
PE2(config)#router ldp
PE2(config-router)#router-id 192.168.0.2
PE2(config-router)#targeted-peer 192.168.0.1
PE2(config-router)#targeted-peer 192.168.0.3
PE2(config-router)#exit
PE2(config)#int vlan 12
PE2(config-if-vlan12)#ldp enable
PE2(config-if-vlan12)#exit
PE2(config)#int vlan 23
PE2(config-if-vlan23)#ldp enable
PE2(config-if-vlan23)#exit

#Configue instance V1 and remote PE in LDP
PE2(config)#vfi v1 100
PE2(config-vfi)#peer 192.168.0.1 pw-class c1
PE2(config-vfi)#peer 192.168.0.3 pw-class c2
PE2(config-vfi)#exit

#Bind VPLS instance mode as VLAN on port, Svid is 200
PE2(config-if-etherne1/0/1)#xconnect vfi 100 mode vlan svid 200

(3) PE3 configuration
It is similar to PE1 and PE2.

5.3.2 Access H-VPLS with LSP

5.3.2.1 Network Requirement

- CE1 and CE2 belong to two stations, while they belong to the same VPN1
- CE1 and CE2 can access to PE via port Eth1/0/1
- N-PE1,N-PE2 and N-PE3 create a full connection VPLS network
- U-PE is an access device of user, it accesses N-PE1 with layered VPLS PW mode

- CE1 can access to U-PE in Ethernet mode, while CE2 can access to N-PE3 in VLAN mode, Svid equals 200.
- Packet encapsulation mode between N-PEs is RAW (Ethernet) mode.
- Connection between U-PE and N-PE1 is PW connection.
- It requires that establish a layer 2 VPN1 by configuring VPLS, making CE1 and CE2 access each other.

5.3.2.2 Group Network Diagram

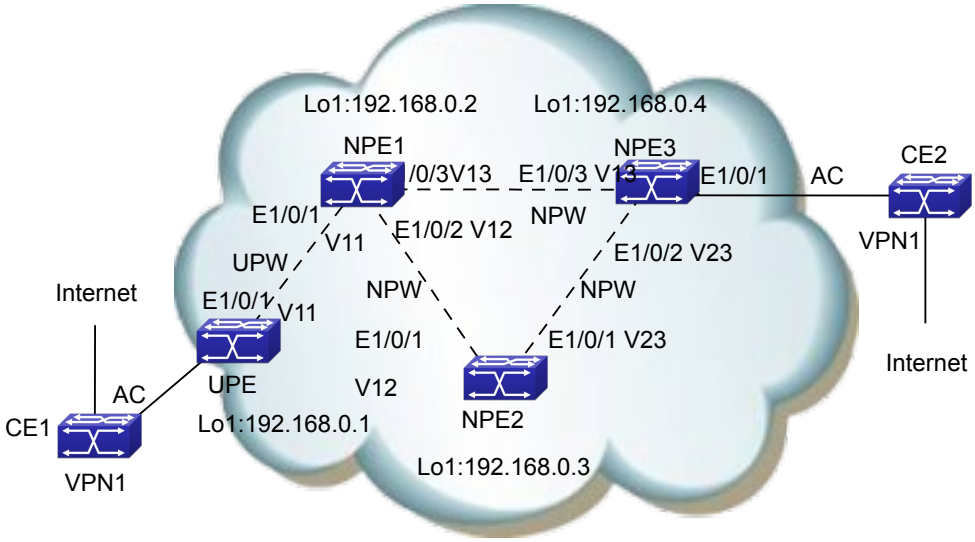

Fig 4-6 H_VPLS network in LSP access mode

5.3.2.3 Configuration Steps

Please configure IP address and mask as shown in the above figure for each interface, including VLAN and Loopback interface. Specific configuration steps are omitted. And this is the configuration only for H-VPLS in LDP.

(1) UPE configuration

#Create PW template

```
UPE(config)#pw-class c1
UPE(config-class)#transport-mode ethernet
UPE(config-class)#exit
```

#Configure the routing

```
UPE(config)#router ospf
UPE(config-router)#router-id 192.168.0.1
UPE(config-router)#network 0.0.0.0/0 area 0
UPE(config-router)#exit
```

#Configure MPLS capability and label switching capability (Interface connected with CE does not enable label switching capability)

```
UPE(config)#mpls enable
UPE(config)#int vlan 11
UPE(config-if-vlan11)#label-switching
UPE(config-if-vlan11)#exit

#Configure LDP
UPE(config)#router ldp
UPE(config-router)#router-id 192.168.0.1
UPE(config-router)#targeted-peer 192.168.0.2
UPE(config-router)#exit
UPE(config)#int vlan 11
UPE(config-if-vlan11)#ldp enable
UPE(config-if-vlan11)#exit

# Configue instance V1 and remote PE in LDP
UPE(config)#vfi v1 100
UPE(config-vfi)#peer 192.168.0.2 pw-class c1
UPE(config-vfi)#exit

#Bind VPLS instance mode as Ethernet on port
UPE(config-if-ethernet1/0/1)#xconnect vfi 100 mode ethernet
```

(2) NPE1 configuration

```
#Create PW template
NPE1(config)#pw-class c1
NPE1(config-class)#transport-mode ethernet
NPE1(config-class)#exit

#Configure the routing
NPE1(config)#router ospf
NPE1(config-router)#router-id 192.168.0.2
NPE1(config-router)#network 0.0.0.0/0 area 0
NPE1(config-router)#exit

# Configure MPLS capability and label switching capability
NPE1(config)#mpls enable
NPE1(config)#int vlan 11
NPE1(config-if-vlan11)#label-switching
NPE1(config-if-vlan11)#exit
NPE1(config)#int vlan 12
NPE1(config-if-vlan12)#label-switching
NPE1(config-if-vlan12)#exit
NPE1(config)#int vlan 13
NPE1(config-if-vlan13)#label-switching
```

```
NPE1(config-if-vlan13)#exit

#Configure LDP
NPE1(config)#router ldp
NPE1(config-router)#router-id 192.168.0.2
NPE1(config-router)#targeted-peer 192.168.0.1
NPE1(config-router)#targeted-peer 192.168.0.3
NPE1(config-router)#targeted-peer 192.168.0.4
NPE1(config-router)#exit
NPE1(config)#int vlan 11
NPE1(config-if-vlan11)#ldp enable
NPE1(config-if-vlan11)#exit
NPE1(config)#int vlan 12
NPE1(config-if-vlan12)#ldp enable
NPE1(config-if-vlan12)#exit
NPE1(config)#int vlan 13
NPE1(config-if-vlan13)#ldp enable
NPE1(config-if-vlan13)#exit
```

#Configue instance V1 and remote PE in LDP (peer in UPE needs to disables level division)

```
NPE1(config)#vfi v1 100
NPE1(config-vfi)#peer 192.168.0.3 pw-class c1
NPE1(config-vfi)#peer 192.168.0.4 pw-class c1
NPE1(config-vfi)#peer 192.168.0.1 no-split-horizon pw-class c1
NPE1(config-vfi)#exit
```

(3) NPE2 configuration

It is similar to NPE1, the difference is no peer, it does not disable level division

(4) NPE3 configuration

#Create PW template

```
NPE3(config)#pw-class c1
NPE3(config-class)#transport-mode ethernet
NPE3(config-class)#exit

#Configure the routing
NPE3(config)#router ospf
NPE3(config-router)#router-id 192.168.0.4
NPE3(config-router)#network 0.0.0.0/0 area 0
NPE3(config-router)#exit

# Configure MPLS capability and label switching capability
NPE3(config)#mpls enable
```

```
NPE3(config)#int vlan 13
NPE3(config-if-vlan11)#label-switching
NPE3(config-if-vlan11)#exit
NPE3(config)#int vlan 23
NPE3(config-if-vlan12)#label-switching
NPE3(config-if-vlan12)#exit

#Configure LDP
NPE3(config)#router ldp
NPE3(config-router)#router-id 192.168.0.4
NPE3(config-router)#targeted-peer 192.168.0.2
NPE3(config-router)#targeted-peer 192.168.0.3
NPE3(config-router)#exit
NPE3(config)#int vlan 13
NPE3(config-if-vlan11)#ldp enable
NPE3(config-if-vlan11)#exit
NPE3(config)#int vlan 23
NPE3(config-if-vlan12)#ldp enable
NPE3(config-if-vlan12)#exit

#Configue instance V1 and remote PE in LDP
NPE3(config)#vfi v1 100
NPE3(config-vfi)#peer 192.168.0.2 pw-class c1
NPE3(config-vfi)#peer 192.168.0.3 pw-class c1
NPE3(config-vfi)#exit

#Bind VPLS instance on port
N-PE3(config-if-ethernet1/0/1)#xconnect vfi 100 mode vlan svid 200
```

5.3.3 Access H-VPLS with QinQ

5.3.3.1 Network Requirement

- CE1 and CE2 belong to two stations, while they belong to the same VPN1
- CE1 and CE2 can access to PE via port Eth1/0/2 and Eth1/0/1 separately
- U-PE is an access device of user, it accesses N-PE1 with layered QinQ mode, Svid is 100
- CE2 uses the normal access mode as VLAN, Svid is 200.
- Packet encapsulation mode between N-PEs is Tagged mode.
- Connection between U-PE and N-PE1 is QinQ connection.
- It requires that establish a layer 2 VPN1 by configuring VPLS, making CE1 and CE2 access each other.

5.3.3.2 Group Network Diagram

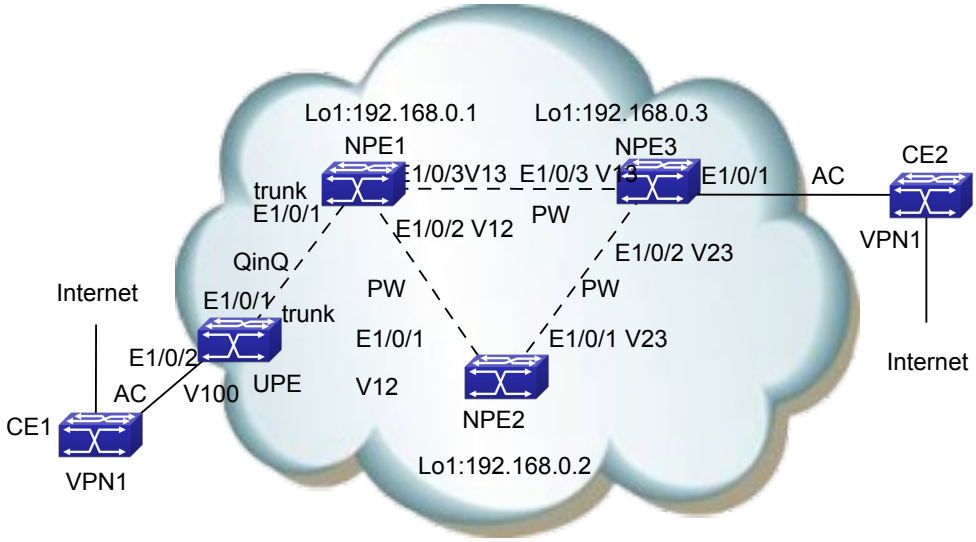

Fig 4-7 H_VPLS network in QinQ access mode

5.3.3.3 Configuration Steps

Please configure IP address and mask as shown in the above figure for each interface, including VLAN and Loopback interface. Specific configuration steps are omitted. And this is the configuration only for H-VPLS in QinQ mode.

(1) UPE configuration

#Enable QinQ on port, VLAN of this port belongs to is 100

UPE(config-if-ethernet1/0/2)#switchport access vlan 100

UPE(config-if-ethernet1/0/2)#dot1q-tunnel enable

#Configure the port connected with NPE1 as Trunk, trunk allows all VLANs to pass by default

UPE(config-if-ethernet1/0/1)#switchport mode trunk

(2) NPE1 configuration

#Create PW template

NPE1(config)#pw-class c1

NPE1(config-class)#transport-mode Vlan

NPE1(config-class)#exit

#Configure the routing

NPE1(config)#router ospf

NPE1(config-router)#router-id 192.168.0.1

NPE1(config-router)#network 0.0.0.0/0 area 0

NPE1(config-router)#exit

Configure MPLS capability and label switching capability
NPE1(config)#mpls enable
NPE1(config)#int vlan 12
NPE1(config-if-vlan12)#label-switching
NPE1(config-if-vlan12)#exit
NPE1(config)#int vlan 13
NPE1(config-if-vlan13)#label-switching
NPE1(config-if-vlan13)#exit

#Configure LDP
NPE1(config)#router ldp
NPE1(config-router)#router-id 192.168.0.1
NPE1(config-router)#targeted-peer 192.168.0.2
NPE1(config-router)#targeted-peer 192.168.0.3
NPE1(config-router)#exit
NPE1(config)#int vlan 12
NPE1(config-if-vlan12)#ldp enable
NPE1(config-if-vlan12)#exit
NPE1(config)#int vlan 13
NPE1(config-if-vlan13)#ldp enable
NPE1(config-if-vlan13)#exit

Configue instance V1 and remote PE in LDP
NPE1(config)#vfi v1 100
NPE1(config-vfi)#peer 192.168.0.2 pw-class c1
NPE1(config-vfi)#peer 192.168.0.3 pw-class c1
NPE1(config-vfi)#exit

#Bind VPLS instance mode as VLAN on port with UPE QinQ access mode, Svid is100
NPE1(config-if-ethernet1/0/2)#switchport mode trunk
NPE1 (config-if-ethernet1/0/2)#xconnect vfi 100 mode vlan svid 100

(3) NPE2 configuration
It is similar to NPE1

(4) NPE3 configuration
#Create PW template
NPE3(config)#pw-class c1
NPE3(config-class)#transport-mode Vlan
NPE3(config-class)#exit

#Configure the routing
NPE3(config)#router ospf
NPE3(config-router)#router-id 192.168.0.3

```
NPE3(config-router)#network 0.0.0.0/0 area 0
NPE3(config-router)#exit

# Configure MPLS capability and label switching capability
NPE3(config)#mpls enable
NPE3(config)#int vlan 13
NPE3(config-if-vlan11)#label-switching
NPE3(config-if-vlan11)#exit
NPE3(config)#int vlan 23
NPE3(config-if-vlan12)#label-switching
NPE3(config-if-vlan12)#exit

#Configure LDP
NPE3(config)#router ldp
NPE3(config-router)#router-id 192.168.0.3
NPE3(config-router)#targeted-peer 192.168.0.1
NPE3(config-router)#targeted-peer 192.168.0.2
NPE3(config-router)#exit
NPE3(config)#int vlan 13
NPE3(config-if-vlan11)#ldp enable
NPE3(config-if-vlan11)#exit
NPE3(config)#int vlan 23
NPE3(config-if-vlan12)#ldp enable
NPE3(config-if-vlan12)#exit

# Configue instance V1 and remote PE in LDP
NPE3(config)#vfi v1 100
NPE3(config-vfi)#peer 192.168.0.1 pw-class c1
NPE3(config-vfi)#peer 192.168.0.2 pw-class c1
NPE3(config-vfi)#exit

#Bind VPLS instance on port
N-PE3(config-if-ethernet1/0/1)#xconnect vfi 100 mode vlan svid 200
```

5.3.4 VPWS Configuration

5.3.4.1 Network Requirement

- CE1 and CE2 belong to two stations, while they belong to the same VPN1
- CE1 and CE2 can access to PE via port Eth1/0/1
- CE1 can access to PE1 in Ethernet mode, while CE2 can access to PE2 in VLAN mode, Svid is 200
- Packet encapsulation mode between PE1 and PE2 is RAW (Ethernet) mode

- It requires to establish a layer 2 VPN1 by configuring VPLS, make CE1 and CE2 can access each other

5.3.4.2 Group Network Diagram

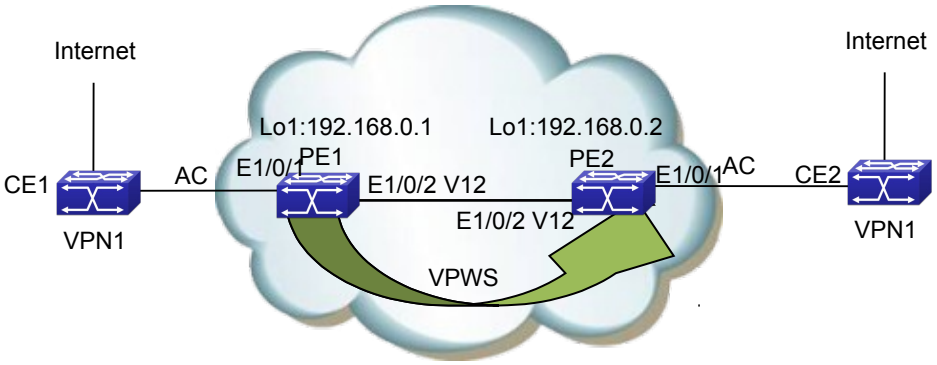

Fig 4-8 VPWS group network diagram

5.3.4.3 Configuration Steps

Please configure IP address and mask as shown in the above figure for each interface, including VLAN and Loopback interface. Specific configuration steps are omitted. And this is the configuration only for VPWS.

(1) PE1 configuration

#Create PW template

PE1(config)#pw-class c1

PE1(config-class)#transport-mode ethernet

PE1(config-class)#exit

#Configuration the routing

PE1(config)#router ospf

PE1(config-router)#router-id 192.168.0.1

PE1(config-router)#network 0.0.0.0/0 area 0

PE1(config-router)#exit

#Configure MPLS capability and label switching capability (Interface connected with CE does not enable label switching capability in non-H-VPLS)

PE1(config)#mpls enable

PE1(config)#int vlan 12

PE1(config-if-vlan12)#label-switching

PE1(config-if-vlan12)#exit

#Configure LDP

PE1(config)#router ldp
PE1(config-router)#router-id 192.168.0.1
PE1(config-router)#targeted-peer 192.168.0.2
PE1(config-router)#exit

#Configure remot PE of VPWS
PE1(config)#l2-vc 192.168.0.2 pw-id 1 pw-class c1

#Bind VPWS mode as ethernet on port
PE1(config-if-ethernet1/0/1)#xconnect l2-vc pw-id 1 mode Ethernet

(2) PE2 configuration
#Create PW template
PE2(config)#pw-class c1
PE2(config-class)#transport-mode vlan
PE2(config-class)#exit

#Configure the routing
PE2(config)#router ospf
PE2(config-router)#router-id 192.168.0.2
PE2(config-router)#network 0.0.0.0/0 area 0
PE2(config-router)#exit

#Configure MPLS capability and label switching capability (Interface connected with CE does not enable label switching capability in non-H-VPLS)
PE2(config)#mpls enable
PE2(config)#int vlan 12
PE2(config-if-vlan12)#label-switching
PE2(config-if-vlan12)#exit

#Configure LDP
PE2(config)#router ldp
PE2(config-router)#router-id 192.168.0.2
PE2(config-router)#targeted-peer 192.168.0.1
PE2(config-router)#exit
PE2(config)#int vlan 12
PE2(config-if-vlan12)#ldp enable
PE2(config-if-vlan12)#exit

#Configure remote PE of VPWS
PE2(config)#l2-vc 192.168.0.1 pw-id 1 pw-class c1

#Bind VPWS mode as VLAN, Svid is 200
PE2(config-if-ethernet1/0/1)#xconnect l2-vc pw-id 1 mode vlan svid 200

5.4 VPLS Troubleshooting

When configure and use VPLS, L2 VPN may not work correctly because of physical connection and error in configuration. So users should note the following factos:

- First, we should ensure that OSPF neighbor among PE1, P and PE2 works correctly, and routing including loopback interface works correctly, that is, check whether all PE devices exist the routing of the peer PE, P.
- Second, we should ensure that whether PE1, P and PE2 globally enable MPLS and LDP or not, at the same time whether they enable Label-switching and LDP at the active interfaces or not. Also we should check whether LDP session on them is correct or not. Furthermore, we should ensure that LDP's remote neighbor, namely targeted peer address, is correct and neighbor is established correctly. At the same time, we should guarantee address of LDP's remote peer is same as that in VFI or L2VC peer.
- Then, in order to check whether PW in PE is established correctly, we can use command **show vpls peer xxx**. The correct PW state is up, or we should check whether PW encapsulation method (Ethernet or vlan), VFI ID, MTU and transmission mode, etc. in two peers are same or not.
- We should judge whether VFI port is bound to correct VFI or not, and whether access mode and Svid are right or not. Additionally if the process is in VPLS mode, we should check whether pw-id bound by port is correct and pw-id bound by PEs are same.
- Finally, in the case of the process are correct, we can use command **show vfi mac-addresses-table** to check MAC address in the local VPN, and both peers can display MAC address of the peer CE.
- Additionally, in the case that MAC address of the peer CE cannot be displayed in CE and flow does not recovered after we restarted devices, please wait for a moment patiently. Because it needs a time to establish connection among OSPF, LDP and PW.
- Notice: please do not enable 802.1x, STP or QinQ, etc. after PE port is bound to VFI. Because this may effect user's network. Furthermore, please do not add this port to layer 3 interface in use of public network, because protocols of routing (for example, OSPF and RIP) and multicast (IGMP and MLD) in layer 3 interface that the port belongs to may result in information leakage of providers to CE.

Chapter 6 MAC-in-MAC

6.1 MAC-in-MAC Overview

MAC-in-MAC, also known as PBB (Provider Backbone Bridge), is defined in IEEE 802.1ah. MAC-in-MAC is a Layer-2 VPN (Virtual Private Network) technique. It encapsulates the customer MAC in the service provider MAC, transmits the inner MAC as payload, and thus improves the expandability for Ethernet and secures services.

6.1.1 Basic Concept of MAC-in-MAC

1. PBBN

A network using MAC-in-MAC is called a provider backbone bridge network (PBBN) or MAC-in-MAC network. For users, a PBBN is a Layer-2 switching network where Layer-2 connections are between different nodes.

2. PBN

A network connecting the PBBN with the customer network is a provider bridge network (PBN). The customer network can connect to the PBBN directly, or through a PBN.

3. MAC-in-MAC packet

A packet encapsulated by MAC-in-MAC is called a MAC-in-MAC packet.

4. BEB

A backbone edge bridge (BEB) is an edge device in the PBBN, like a PE device in an MPLS network. The BEB encapsulates packets from the customer network by using MAC-in-MAC and forwards them to the PBBN, or de-encapsulates MAC-in-MAC packets from the PBBN and forwards them to the customer network.

5. BCB

A backbone core bridge (BCB) is a core device in the PBBN, like a P device in an MPLS network. It forwards MAC-in-MAC packets according to B-MAC and B-VLAN. A BCB device only forwards packets and learns MAC addresses in the backbone network. It does not learn a large number of customer MAC addresses. In this way, the network deployment costs are reduced, and the PBBN is given better expandability.

6. B-MAC/B-VLAN

When encapsulating a customer packet, a BEB tags the packet with the service provider MAC address (known as backbone MAC address, B-MAC) and service provider

VLAN (known as backbone VLAN, B-VLAN). Note that the B-MAC falls into source B-MAC and destination B-MAC. In the PBBN, a BCB forwards MAC-in-MAC packets according to their B-MAC and B-VLAN.

7. Uplink port/downlink port

The port that connects the BEB to the PBBN is the uplink port, and the port that connects the BEB to the customer network is the downlink port. After the packets from the customer network are encapsulated in MAC-in-MAC packets, they are forwarded out from the corresponding uplink ports on the BEB; after the MAC-in-MAC packets from the PBBN are de-encapsulated, they are forwarded out from the corresponding downlink port on the BEB according to the customer MAC.

8. MAC-in-MAC instance and I-SID

In the PBBN, a MAC-in-MAC instance represents a type of services provided by the service provider, and is uniquely identified by a backbone service instance identifier (I-SID).

6.1.2 Basic Network Model of MAC-in-MAC

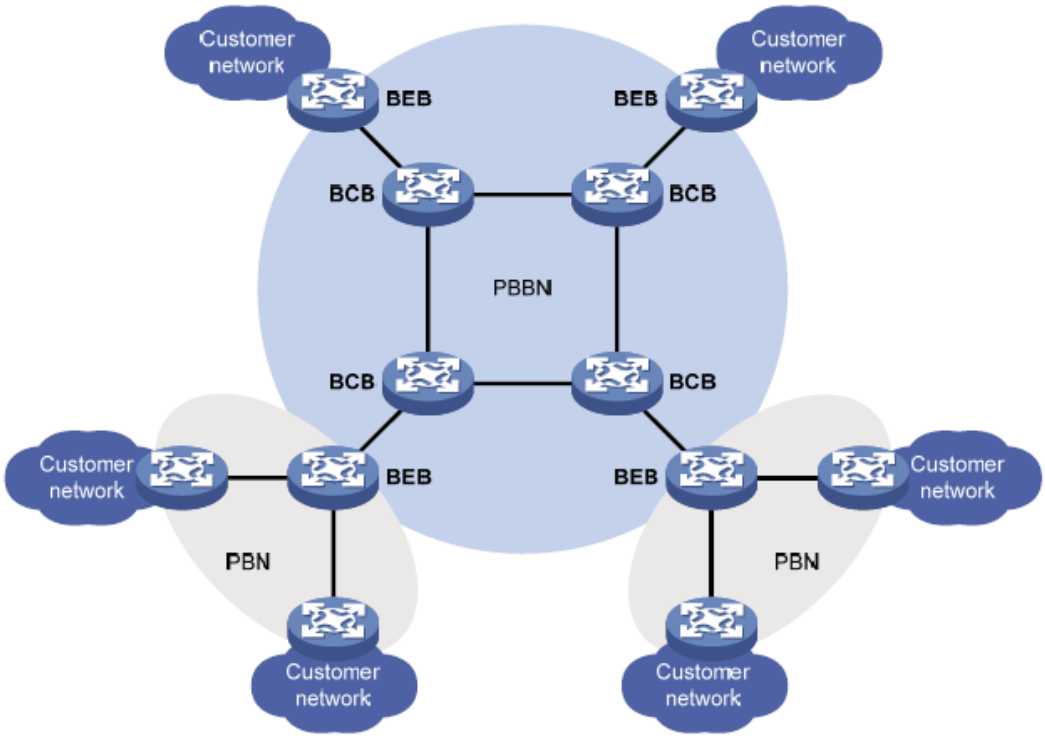

Fig 0-1 Basic Network Model of MAC-in-MAC

MAC-in-MAC model consists of CE, BEB and BCB, the basic concepts are shown in the following.

- **CE** (Custom Edge)

Custom Edge device connects with the service provider directly. It may be a router, switch or a host.

- **BEB**

A backbone edge bridge (BEB) is an edge device in the PBBN, like a PE device in an MPLS network. The BEB encapsulates packets from the customer network by using MAC-in-MAC and forwards them to the PBBN, or de-encapsulates MAC-in-MAC packets from the PBBN and forwards them to the customer network.

- **BCB**

A backbone core bridge (BCB) is a core device in the PBBN, like a P device in an MPLS network. It forwards MAC-in-MAC packets according to B-MAC and B-VLAN. A BCB device only forward packets and learn MAC addresses in the backbone network. It does not learn a large number of customer MAC addresses. In this way, the network deployment costs are reduced, and the PBBN is given better expandability

6.1.2.1 Basic Network Model of H-PBBN

H-PBBN（Hierarchical Provider Backbone Bridging Network） enables a PBB network with the lower level to pass through in a PBB network with the higher level, it encapsulate a outer MIM header for MAC-in-MAC packets in a PBB network with the lower level to form double-layer nesting about MIM header

Fig 0-2 Basic Model of H-PBBN

6.1.3 Packet Encapsulation of MAC-in-MAC

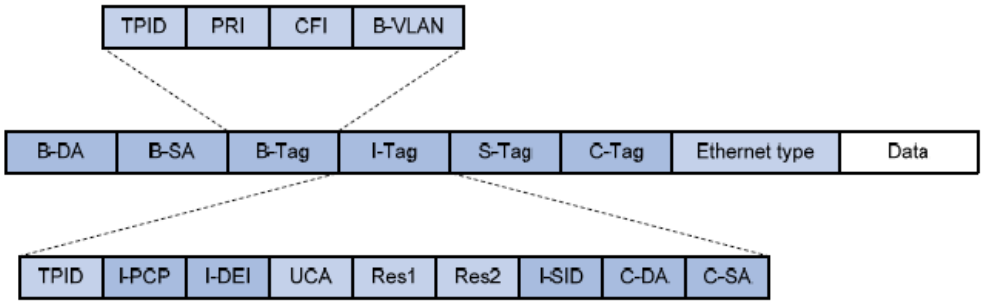

Fig 0-3 Packet encapsulation packet of MAC-in-MAC

Field	Full name	Description
B-DA	Backbone Destination MAC address	Destination B-MAC, outer destination MAC address in a MAC-in-MAC packet. It is the MAC address of the BEB device at the destination end of the PBBN tunnel. The combination of B-DA and B-SA is B-MAC.
B-SA	Backbone Source MAC address	Source B-MAC, outer source MAC address in a MAC-in-MAC packet. It is the MAC address of the BEB device at the source end of the PBBN tunnel. The combination of B-DA and B-SA is B-MAC.
B-Tag	Backbone VLAN Tag	B-VLAN Tag, outer VLAN tag in a MAC-in-MAC packet. It indicates the VLAN information and priority information of the packet within the PBBN. The Tag Protocol Identifier (TPID) in the B-tag is 0x8100.
I-Tag	Backbone Service Instance Tag	Service identifier of a MAC-in-MAC packet. The I-tag contains the backbone service instance priority code point (I-PCP) and backbone service instance drop eligibility indicator (I-DEI) on the BEB, backbone service instance identifier (I-SID), and the C-DA and C-SA of the customer packet. The TPID of the I-tag is 0x88E7.
S-Tag	Service VLAN Tag	Outer VLAN tag of the packet in the PBN, which indicates the VLAN information and priority information of the packet within the PBN.
C-Tag	Customer VLAN Tag	Inner VLAN tag of the packet in the PBN, which indicates the VLAN information and priority information of the packet within the customer network.

Table 0-1 Some key fields of a MAC-in-MAC packet encapsulation format

6.1.4 Packet Forwarding of MAC-in-MAC

Fig 0-4 PBBN network application

6.1.4.1 Access Mode of Downlink Port

There are two access modes on downlink port in the basic model of PBBN, they are VLAN and ethernet, descriptions are shown in the following:

- Access mode of VLAN: Ethernet frame header sent by CE to BEB or BEB to CE with a VLAN TAG, it is a service delimiter added by provider's device to distinguish users, and we call it as S-TAG.
- Access mode of Ethernet: Ethernet frame header sent by CE to BEB or BEB to CE without a service delimiter. If there is a VLAN TAG in it, however, this TAG is an inner one, and it is meaningless to BEB. We call it as C-TAG.

6.1.4.2 Packet Forwarding of Ethernet Access

In Fig 0-4, downlink port of BEB uses the access mode of Ethernet, packet forwarding procedures from customer A to customer B are as follows:

1. Customer A sends the packets with U-Tag to BEB1.
2. BEB1 chooses an appropriate B-DA according to the destination MAC address and adds an outer MAC header for the packets.
3. In order to transmit the packets via the public network tunnel, BEB1 adds the public network B-VLAN for the outer MAC header and transmit the packets to BEB2.
4. After BEB2 receives the packets, it finds out the relevant VFI according to B-SA+B-VLAN, and sends the packets with U-TAG to customer B.

6.1.4.3 Packet Forwarding of VLAN Access

In Fig 0-4, downlink port of BEB uses the access mode of VLAN, packet forwarding procedures from customer A to customer B are as follows:

1. Packets sent by customer A should be with S-TAG. If packets forwarded by CE1 without VLAN TAG or with unmatched VLAN TAG, it executes the normal layer-2 forwarding process.
2. BEB1 chooses an appropriate B-DA according to the destination MAC address and

adds an outer MAC header for the packets.
3. In order to transmit the packets via the public network tunnel, BEB1 adds the public network B-VLAN for the outer MAC header and transmit the packets to BEB2.
4. After BEB2 receives the packets, it finds out the relevant VFI according to B-SA+B-VLAN, and sends the packets with S-TAG to customer B.

6.1.5 MAC-in-MAC Advantages

MAC-in-MAC advantages are shown in the following:

- MAC-in-MAC solves the limit about QinQ only with 4096 SVLANs.
- MAC-in-MAC isolates the MAC of the customers, and enhances the security of Ethernet service.
- MAC-in-MAC extends the space of the MAC addresses.

6.2 MAC-in-MAC Configuration

MAC-in-MAC configuration task list:

1. Configure VFI and the relevant parameters (required)
   (1) Create VFI, specify VFI ID and ISID and enter VFI view
   (2) Configure BVLAN for VFI (optional)
   (3) Configure the default multicast group for VFI (optional)
2. Configure downlink port to bind with VFI (required)
   (1) Enter port view
   (2) Bind downlink port to the specified VFI and enter the specified access mode
3. Configure uplink port to bind with VFI (required)
   (1) Enter port view
   (2) Configure a port to be uplink port of VFI

1. Configure VFI and the relevant parameters (required)
   (1) Create VFI, specify VFI ID and ISID and enter VFI view
   (2) Configure BVLAN for VFI (optional)
   (3) Configure the default multicast group for VFI (optional)

Command	Explanation
Global mode	
vfi <*vfi-name*> <*vfi-id*> mim <*i-sid*> **no vfi <*vfi-name*>**	Create a MIM instance and enter VFI mode, at the same time, we must specify the unique MIM instance name, VFI-ID and ISID in global mode. No command deletes the corresponding VFI.
VFI mode	
mim bvlan <*vlan-id*> **no mim bvlan <*vlan-id*>**	Specify a BVLAN for VFI instance, no command restores the default BVLAN.

mim address destination default <*mac-addr*> **no mim address destination default <*mac-add*>**	Specify the remote destination B-DA, no command restores the default value.

2. Configure downlink port to bind with VFI (required)

(1) Enter port view

(2) Bind downlink port to the specified VFI and enter the specified access mode

Command	Explanation
Port mode	
xconnect vfi <*vfi-id*> [mode {ethernet | vlan [svid <*svid*>]}] **no xconnect vfi <*vfi-id*> [mode {ethernet | vlan [svid <*svid*>]}]**	Bind a downlink port with a MIM instance and enter the access mode. No command deletes the MIM instance bound by the port.

3. Configure uplink port to bind with VFI (required)

(1) Enter port view

(2) Configure a port to be uplink port of VFI

Command	Explanation
Port mode	
mim uplink vfi <*vfi-id*> **no mim uplink vfi <*vfi-id*>**	Specify a port to be the uplink port of MAC-in-MAC instance, no command cancels the operation.

6.3 Typical Example of MAC-in-MAC

6.3.1 Basic Application Scene of MAC-in-MAC

6.3.1.1 Network Requirement

- ☞ CE1 and CE2 belong to two stations of CustomerA and CustomerB respectively, and they belong to the same VPN1.
- ☞ CE1 and CE2 are connected to BEB devices via Eth1/0/1 respectively.
- ☞ Access mode of Ethernet is used between CE1 and BEB1, access mode of VLAN is used between CE2 and BEB2, and Svid is set as 200.
- ☞ Create layer-2 VPN1 by configuring MAC-in-MAC mode, which enable CE1 and CE2 to access each other.

6.3.1.2 Network Diagram

Fig 0-5 PBBN network application

6.3.1.3 Configuration Procedures

(1) BEB1 configuration
#Create VFI
BEB1(config)#vfi a 1 mim 17
BEB1(config-vfi)#mim bvlan 2
BEB1(config-vfi)#exit
BEB1(config)#

#Configure the downlink port
BEB1(config)#interface ethernet 1/0/1
BEB1(config-if-ethernet1/0/1)#xconnect vfi 1
BEB1(config-if-ethernet1/0/1)#exit
BEB1(config)#

#Configure the uplink port
BEB1(config)#interface ethernet 1/0/2
BEB1(config-if-ethernet1/0/2)#switchport mode trunk
BEB1(config-if-ethernet1/0/2)#mim uplink vfi 1
BEB1(config-if-ethernet1/0/2)#exit
BEB1(config)#

(2) BEB2 configuration
#Create VFI
BEB1(config)#vfi a 1 mim 17
BEB1(config-vfi)#mim bvlan 2
BEB1(config-vfi)#exit
BEB1(config)#

#Configure the downlink port
BEB1(config)#interface ethernet 1/0/1

```
BEB1(config-if-ethernet1/0/1)#xconnect vfi 1 mode vlan svid 200
BEB1(config-if-ethernet1/0/1)#exit
BEB1(config)#

#Configure the uplink port
BEB1(config)#interface ethernet 1/0/2
BEB1(config-if-ethernet1/0/2)#switchport mode trunk
BEB1(config-if-ethernet1/0/2)#mim uplink vfi 1
BEB1(config-if-ethernet1/0/2)#exit
BEB1(config)#
```

6.4 MAC-in-MAC Troubleshooting

When configuring and using MAC-in-MAC, L2 VPN may be abnormity due to the reasons, such as physical connection and false configuration, please pay attention to the following problems.

- First, ensure that physical connection is normal between BEB1, BCB and BEB2.
- Secondly, ensure that ISIDs of the different BEB are same within a VPN when creating VFI, or else communication is failing.
- Then, ensure that uplink port (it must be trunk or hybrid port) and downlink port are added to VFI of BEB.
- Ensure that the configured BVLAN and the native vlan of uplink port must be different in VFI, or else communication is failing.
- If downlink port uses the access mode of VLAN, we should ensure that the tag of the packets received by BEB accords with svid of downlink port, or else communication is failing.
- Ensure that downlink port with tag is added to BVLAN, or else it will result in the packets without tag from downlink port, so as to communication is falling.
- Finally, in the case of the process are correct, we can use command show vfi mac-addresses-table to check MAC address of the local VFI on BEB1 and BEB2, and both peers can show MAC address of the peer CE.