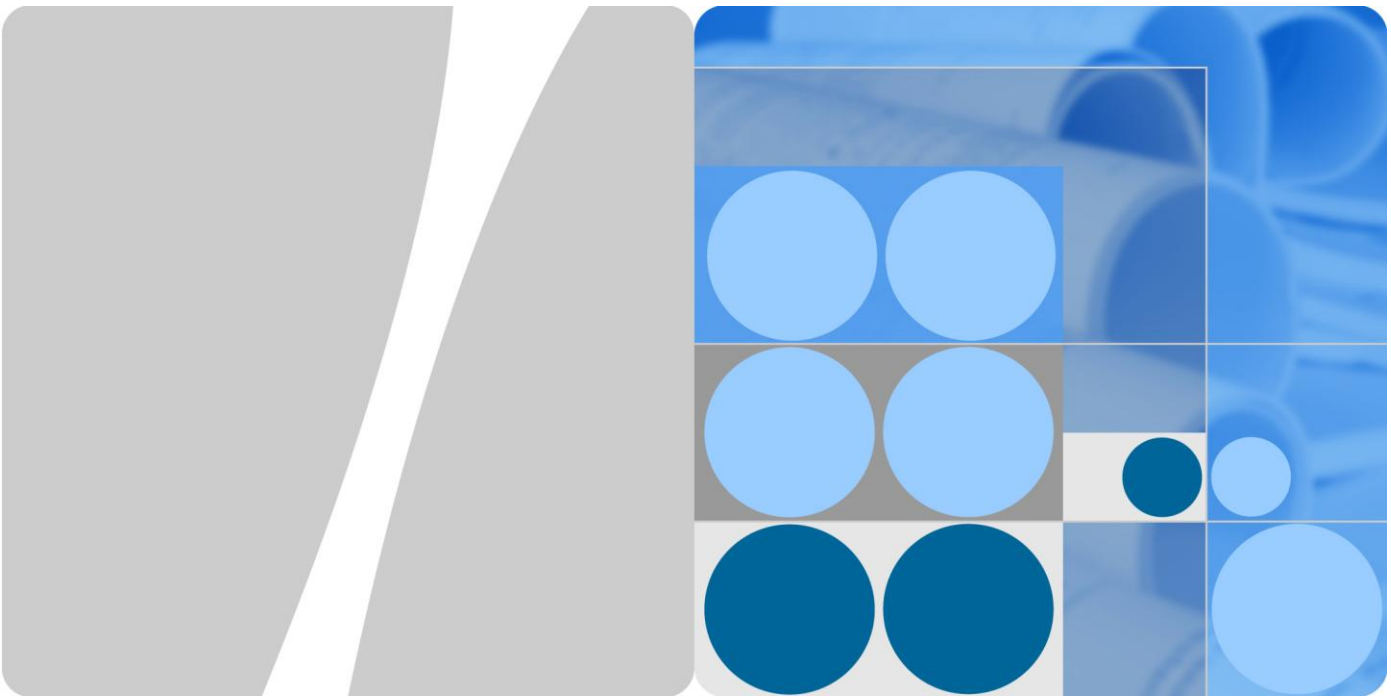


Technology White Paper - TRILL

Issue 01

Date 2013-3-31

HUAWEI TECHNOLOGIES CO., LTD.

Trademarks and Permissions

 and other Huawei trademarks are trademarks of Huawei Technologies Co., Ltd.

All other trademarks and trade names mentioned in this document are the property of their respective holders.

Notice

The purchased products, services and features are stipulated by the contract made between Huawei and the customer. All or part of the products, services and features described in this document may not be within the purchase scope or the usage scope. Unless otherwise specified in the contract, all statements, information, and recommendations in this document are provided "AS IS" without warranties, guarantees or representations of any kind, either express or implied.

The information in this document is subject to change without notice. Every effort has been made in the preparation of this document to ensure accuracy of the contents, but all statements, information, and recommendations in this document do not constitute the warranty of any kind, express or implied.

Huawei Technologies Co., Ltd.

Address: Huawei Industrial Base
Bantian, Longgang
Shenzhen 518129
People's Republic of China

Website: http://enterprise.huawei.com

Email: ChinaEnterprise_TAC@huawei.com

Contents

Figures

1 TRILL

1.1 Introduction to TRILL

Definition

Transparent Interconnection of Lots of Links (TRILL) is an Internet Engineering Task Force (IETF) protocol standard that uses Layer 3 routing techniques on Layer 2 networks. TRILL implements Layer 2 routing by extending Intermediate System to Intermediate System (IS-IS) to meet requirements of large Layer 2 networking in data centers and to provide solutions for data center services.

Purpose

In the cloud computing era, a data center stores, queries, and searches for mass data using the distributed architecture. In the data center, a huge amount of collaboration is required between servers for CSS calculation, generating a high volume of east-to-west traffic between servers. CSS calculation is achieved by virtualization technologies, improving the computation capabilities of devices. For example, server virtualization improves several times the throughput of physical servers. To improve service reliability, reduce IT costs and operation and maintenance costs, and increase service deployment flexibility, virtual machines (VMs) must be able to dynamically migrate within a data center.

To meet the preceding service requirements of data centers, the architecture of data centers must support the following functions:

- Smooth VM migration

  As one of core cloud computing technologies, server virtualization has been widely used. To maximize service reliability, reduce IT costs and operation and maintenance costs, and increase service deployment flexibility in a data center, VMs must be able to dynamically migrate within the data center instead of being restricted to an aggregation or access switch.
- Non-blocking, low-delay data forwarding

  Different from traditional carrier traffic models, most traffic of a data center is east-to-west traffic between servers. To ensure service provisioning, non-blocking, low-delay data forwarding is required. Currently, the fat-tree topology is a widely accepted non-blocking network architecture in the industry.
- Multitenant

In the cloud computing era, a physical data center is shared by multiple tenants instead of being exclusively used by one tenant. In this situation, each tenant corresponds to a virtual data center instance to use exclusive servers and storage as well as network resources. To ensure security, data traffic between tenants needs to be isolated. In traditional Layer 2 networking, the number of supported tenants is limited by the number of VLANs, which is at most 4096. As cloud computing technologies develop, the number of tenants supported by the future data center network architecture must exceed 4096.

- Large-capacity, scalable network

  In the cloud computing era, a large data center must support hundred thousands or millions of servers. To implement non-blocking data forwarding, several hundreds or thousands of switches are required for a large data center. In such a large-scale network, loop prevention protocols must be configured. When a fault occurs on a network node or link, fast network convergence can be triggered to recover services rapidly. Network maintenance and service deployment are simple. In addition, data center networks must be highly scalable to meet rapid development of data centers.

The traditional network layers of Layer 2 access and Layer 3 aggregation/core cannot meet the preceding requirements for network architecture of data centers. Therefore, the large Layer 2 fat-tree architecture is widely deployed. To deploy non-blocking networks, implement smooth VM migration, and adapt to network scale expansion, TRILL was introduced. Compared to traditional Layer 2 xSTP protocols and Layer 3 routing protocols, TRILL has the following advantages:

- Efficient, non-blocking forwarding

  On a TRILL network, each device regards itself as the source node to calculate the shortest path to all other nodes through the shortest path tree (SPT) algorithm. If multiple equal-cost links are available, load balancing can be implemented when unicast forwarding entries are generated. In data center fat-tree networking, forwarding data along multiple paths can maximize network bandwidth efficiency and implement non-blocking forwarding. Traditional xSTP protocols can implement forwarding only through a single path by blocking links, which wastes bandwidth and conflicts with non-blocking networking.

  On a TRILL network, equal cost multipath (ECMP) and SPT can be used for data packet forwarding. Therefore, TRILL networking can greatly improve the data forwarding efficiency of data centers and increase the data center network throughput.

- Smooth VM migration

  On a data center network, VMs must be able to dynamically migrate within the data center. To ensure proper running of services, IP addresses and MAC address of VMs must remain the same before and after the migration. On a Layer 2 xSTP aggregation+Layer 3 IP routing network, IP addresses of VMs are changed if VMs migrate across the network segment. Deployed on a large Layer 2 network, TRILL supports dynamic VM migration within the data center.

- Loop prevention

  TRILL can automatically select the root of a distribution tree, allowing each Routing Bridge (RBridge) node to use the root as the source node to calculate the shortest paths to all other RBridges. In this manner, the multicast distribution tree to be shared by the entire network can be automatically built. This distribution tree connects all nodes on the network to each other and prevents loops when Layer 2 unknown unicast, multicast, or broadcast data packets are transmitted on the network.

- Fast convergence

  On a traditional Layer 2 network, an Ethernet header does not carry the TTL field, and the xSTP convergence mechanism is not well designed. As a result, when the network

topology changes, network convergence is slow and may even last tens of seconds, failing to provide high service reliability for data centers. TRILL uses routing protocols to generate forwarding entries, and the TRILL header carries the Hop-Count field that allows temporary loops. These advantages allow for sub-second network convergence when a fault occurs on a network node or link.

- Flexible deployment

  The TRILL protocol is easy to configure because many configuration parameters such as **Nickname** and **systemID** can be automatically generated and many protocol parameters can retain default settings. On a TRILL network, unicast and broadcast protocols are managed in a unified manner. This is different from the situation on a Layer 3 network where multiple sets of routing protocols such as IGP and PIM need to be separately maintained for unicast routing and multicast routing. Moreover, a TRILL network is still a Layer 2 network, which has the same characteristics as a traditional Layer 2 network: plug-and-play and ease of use.

- Easy support of multitenant

  Currently, TRILL uses the VLAN ID as the tenant ID and isolates tenant traffic using VLANs. A maximum of 4096 VLAN IDs will not become a bottleneck in the initial phase of the cloud computing industry and large Layer 2 network operation. As the cloud computing industry develops, TRILL must break through the limitation of 4096 tenant IDs. To address this issue, TRILL will use FineLable to identify tenants. FineLable is 24 bits long and supports a maximum of 16M tenants, which can meet tenant scalability requirements.

- Smooth evolution of large network

  A large Layer 2 network with TRILL deployed supports around 1000 switches. Traditional Layer 2 networks using the xSTP protocols can seamlessly connect to a TRILL network. Servers connected to xSTP networks can communicate with servers connected to a TRILL network at Layer 2, and VMs can migrate within the TRILL network.

Non-Blocking Network and Fat-Tree Topology

Service and resource planning requirements of cloud-computing data centers differ significantly from those of traditional data centers. These requirements bring great changes to data center networks. Change into the traffic model is the most significant change and brings new challenges to data center networks. It is estimated that about 70% of traffic in a cloud-computing data center is east-west traffic, while about 80% of traffic in a traditional data center is north-south traffic.

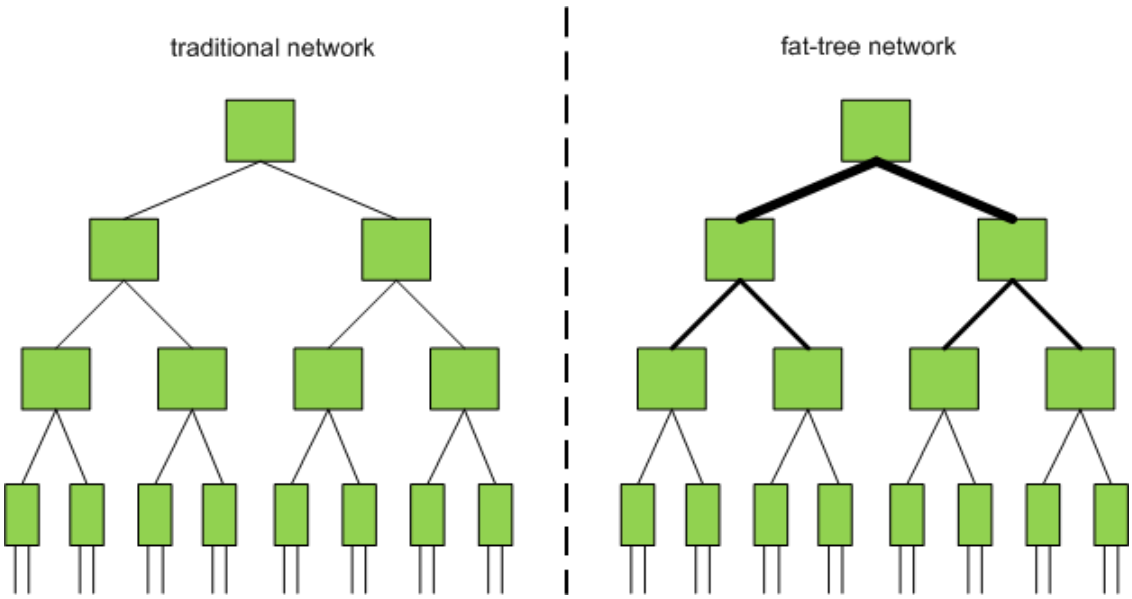
Traditional data centers provide access for external users; therefore, most traffic moves in a north-south direction. Based on services characteristics and limits on egress bandwidth, bandwidth is allocated to each layer with a specified oversubscription ratio between layers. Bandwidth on the access layer is several times that on the aggregation or core layer. The common oversubscription ratio ranges from 1:3 to 1:20.

The increasing variety of services brings great challenges to the data center traffic model. For example, bandwidth-intensive services such as searching and parallel computing require clusters constituted by a large number of servers. Collaboration between servers results in a sharp increase in east-west traffic between servers. Dynamic VM migration makes the traffic model more complexity, and east-to-west traffic becomes a majority in data centers.

Traditional data center networks are unable to handle traffic in the new traffic model. A data center requires a non-blocking network with equal bandwidth on each layer for line-rate traffic transmission between servers.

Currently, the fat-tree topology is a widely accepted non-blocking network architecture in the industry. This architecture uses a large number of low-performance switches to construct a large non-blocking network. In a traditional tree topology, bandwidth is allocated on each layer with a specified oversubscription ratio between layers. The amount of bandwidth at the root is much less than the total bandwidth of all the leaves. The fat-tree architecture looks more like a real tree. The root has higher bandwidth than leaves. This fat-tree architecture is the basis of a non-blocking network. Figure 1-1 shows the topologies of a traditional network and a fat-tree network.

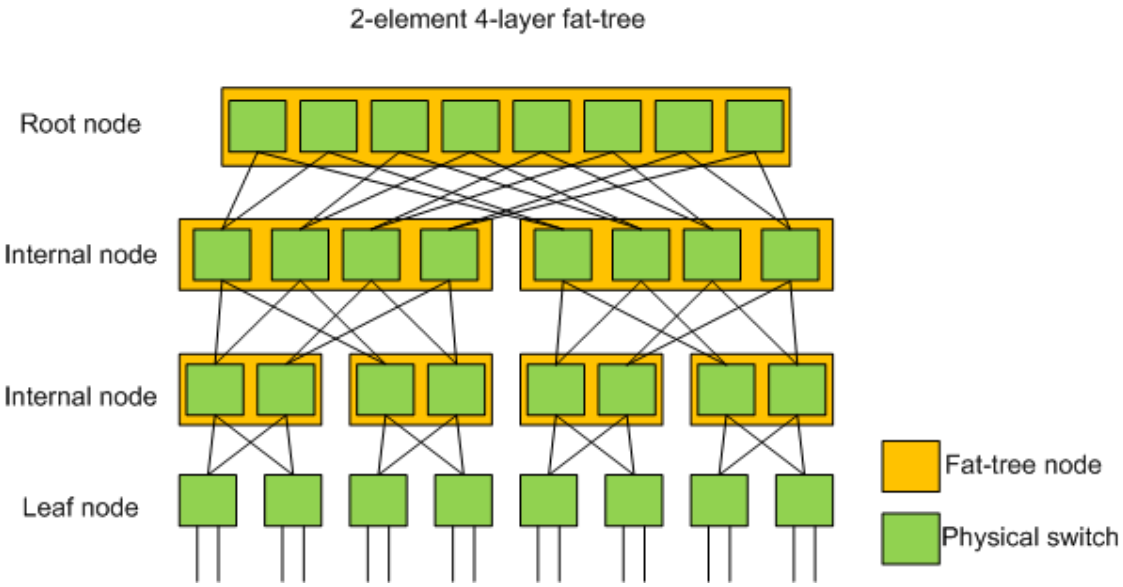
Figure 1-1 Comparison between a traditional network and a fat-tree network



On the fat-tree network, each node excluding the root has the same uplink bandwidth and downlink bandwidth. All nodes on the tree support line-rate forwarding.

Figure 1-2 shows a 2-element 4-layer fat-tree architecture.

Figure 1-2 Fat-tree physical topology



Each leaf node represents a switch connecting to two terminals. Two switches constitute a logical node at layer 2 and four switches constitute a logical node at layer 3. The logical nodes at layer 2 and layer 3 are called internal nodes. Eight switches constitute the logical root. A logical node is a virtual device formed by multiple switches or a device with forwarding capabilities of multiple switches. The uplink bandwidth and the downlink bandwidth on each logical node are the same. There is no bandwidth oversubscription on the network.

Only a half of uplink bandwidth on the root is used for downstream access, and the other half of uplink bandwidth is retained for scalability. The network can be expanded by extending the fat tree towards the root.

Benefits

TRILL brings the following benefits to data center operators:

- Enables large Layer 2 data centers to implement non-blocking VM migrations, simplifying network management.
- Allows TRILL devices to seamlessly connect to devices enabled with traditional bridging functions, lowering network upgrade costs.

1.2 References

The following table lists the references.

Document	Description	Remarks
RFC 6325	TRILL Base Protocol Specification	TRILL supports a maximum of two multicast trees on the entire network. One RB only advertises one nickname.
RFC 6327	TRILL Adjacency	-
RFC 5556	TRILL Problem and Applicability Statement	-
draft-ietf-trill-rbridge-af-04	RBridges: Appointed Forwarders	-
draft-ietf-trill-rbridge-channel-01	RBridges: TRILL RBridge Channel Support	-

1.3 Availability

Version Support

Product	Version
CE12800&CE5800&CE6800	V100R001C00

Feature Dependency

Dependency between TRILL and other features is as follows:

- TRILL depends on VLAN.
- After TRILL is enabled in a VLAN, the VLAN cannot be changed.

Hardware Requirements

The TRILL feature does not require additional hardware.

Specifications

Item	CE12800	CE6800&CE5800
Maximum number of neighbors that can be established on an interface on a non-broadcast network	8	8
Maximum number of neighbors that can be established on the device	256	32
Maximum number of neighbors that can	200	200

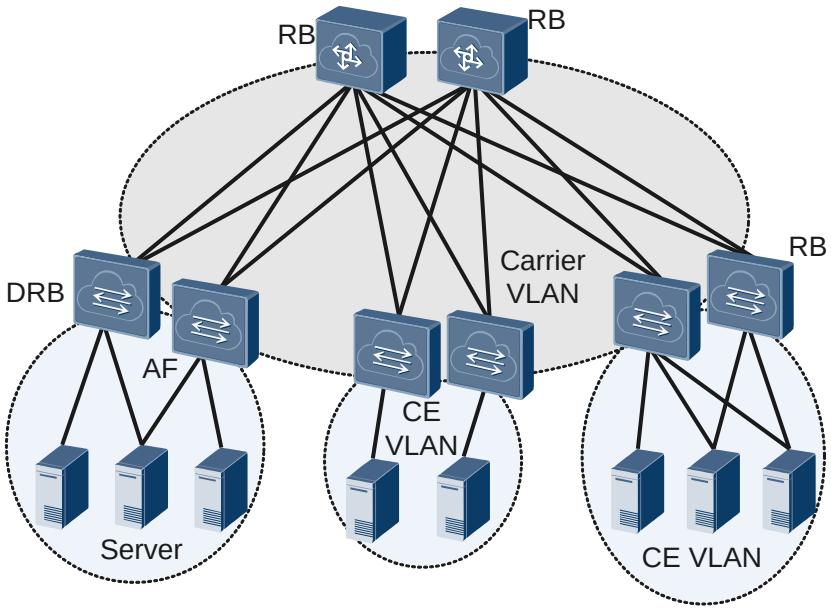
Item	CE12800	CE6800&CE5800
be established on an interface on a broadcast network		

1.4 Principles

1.4.1 Basic Concepts

This section introduces basic concepts about Transparent Interconnection of Lots of Links (TRILL). Figure 1-3 shows basic roles in the typical TRILL networking.

Figure 1-3 Large-scale Layer 2 TRILL networking



Devices in TRILL Networking

RB

A routing bridge (RB) is a Layer 2 switch that runs TRILL.

DRB

A designated routing bridge (DRB) is an RB that functions as a transit device and performs special tasks on TRILL networks. On a TRILL broadcast network, if two RBs are located on the same virtual local area network (VLAN), the RB whose interface with a higher DRB priority or larger MAC address is selected as the DRB when they are establishing neighbor relationships. The DRB communicates with each device on the network to synchronize all the link state databases (LSDBs) on the VLAN, sparing every two devices from communicating for LSDB synchronization. DRBs perform the following tasks:

- Generate pseudonode link state protocol data units (LSPs) when more than two RBs exist on the network.
- Send complete sequence number protocol data units (CSNPs) to synchronize LSDBs.
- Select an carrier VLAN as the Designated VLAN, the DVLAN will transmit user packets and TRILL control packets.
- Select the appointed forwarder (AF). Only one RB can function as the AF for a customer edge (CE) VLAN.

AF

An AF is the RB that is selected by a DRB to transmit user packets. Only AFs can transmit user packets.

VLANs on a TRILL Network

Carrier VLAN

A physical local area network (LAN) is divided into several logical broadcast domains (VLANs). Devices can communicate with each other only within a VLAN. Therefore, transmission of broadcast packets is confined to one VLAN to ensure the security of local area network.

Carrier VLANs transmit TRILL data and protocol packets, not Ethernet data packets. A maximum of three carrier VLANs can be configured on one RB.

CE VLAN

A CE VLAN, also referred to as access Carrier VLAN, is used for CE users to access TRILL networks. CE VLANs transmit Ethernet data packets only.

Designated VLAN

The designated VLAN (DVLAN) is the VLAN that transmits TRILL data and control packets.

Nickname

Each RB on a TRILL network has a unique nickname. The nickname is similar to an IP address in terms of function.

A nickname has one priority and one root priority.

- When a nickname conflict occurs on a TRILL network, the priority determines which RB's nickname is to be advertised to other RBs.

1. The RB with the highest priority advertises its nickname.
2. If the RBs with the same nickname have the same priority, the RB with the largest system ID advertises its nickname.

- An RB uses its root priority to run for the root of multicast tree. The RBs with the highest and second-highest root priority are selected as the roots of two multicast trees.

Interface Roles

Interfaces of switches on TRILL networks are classified into the following types:

- Trunk interfaces: connect switches and transmit TRILL data packets and protocol packets only. P2P interfaces are special trunk interfaces that connect switches. The switches connected by P2P interfaces cannot be elected as a DRB.
- Access interfaces: transmit Native Ethernet packets and protocol packets only.
- Hybrid interfaces: transmit both TRILL data and protocol packets and Native Ethernet packets by default.

By default, the type of TRILL interfaces is trunk.

TRILL Address Structure

TRILL and IS-IS both use network service access point (NSAP) addresses. An NSAP address, such as 00.1234.5678.9abc.00, contains the following parts:

- Area ID: An area ID identifies an area. An IS-IS network has multiple areas, while a TRILL network has only one area. The TRILL area ID is 00.
- System ID: A 48-byte system ID uniquely identifies a host or switch.

  In actual applications, System ID can be automatically generated or manually configured. The automatically generated system ID is the same as the bridge MAC address of the RB. If you want to configure a system ID, make sure that it is unique on the whole network.
- SEL (also referred to as NSAP Selector or N-SEL): The role of a SEL is similar to that of the protocol identifier of IP. Each transport protocol has one unique SEL. The SEL of TRILL is 00.

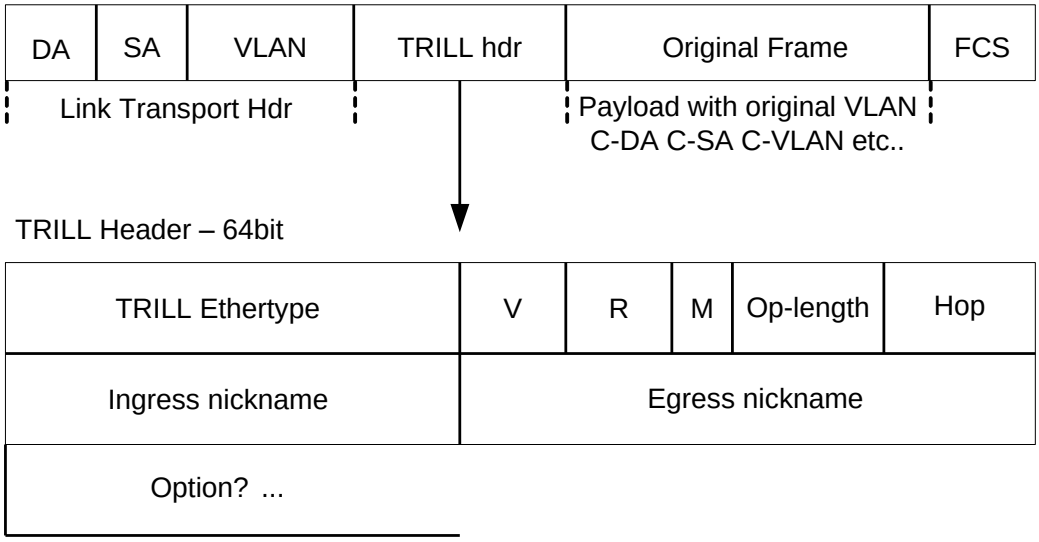
NET

A network entity title (NET) indicates a switch's network layer information and can be viewed as a special NASP address. For example, if a NET is 00.1234.5678.9abc.00, 1234.5678.9abc is the system ID, the first 00 is the area ID, and the ending 00 is the SEL.

TRILL Packet Encapsulation Mode

TRILL packets are encapsulated in the MAC-in-MAC mode. In the public area of the TRILL network, data packets pass through the traditional bridge and hub and are forwarded based on outer Ethernet headers. Figure 1-4 shows the header of a TRILL packet.

Figure 1-4 TRILL packet header



Each field is described as follows:

- DA: Outer destination MAC address.
- SA: Outer source MAC address.
- VLAN: Outer VLAN ID.
- V: TRILL version. The current value is 0. If V is not 0, the packet is discarded.
- M: Whether the packet is a multicast packet. 0 indicates a known unicast packet; 1 indicates an unknown unicast, multicast, or broadcast packet. When the value of M is 1, **Egress RBridge** indicates the root of the multicast tree that is forwarding packets.
- Op-Length: Length of the TRILL extension header (Option field).
- Egress RBridge Nickname: Nickname of the egress device of the TRILL network. The Egress RBridge Nickname in a known unicast packet identifies the RBridge specified by the private destination MAC address; the Egress RBridge Nickname in a multicast packet identifies the root RBridge of the multicast tree. The value of this field cannot be changed on an intermediate RBridge.
- Ingress RBridge Nickname: Nickname of the ingress device of the TRILL network. This field identifies the first edge RBridge from which packets enter the TRILL area. The value of this field cannot be changed on an intermediate RBridge.
- Options: Whether reserved content is identified and processed by each hop or only by the first and end nodes. This field defines only two 1-bit flags: Critical Hop by Hop (CHbH) and Critical Ingress to Egress (CItE). The actual Option is not defined currently.

1.4.2 TRILL Mechanisms

Routing bridges (RBs) on a Layer 2 network can communicate with each other only after the following tasks are complete.

- TRILL Neighbor Relationship Establishment
- LSDB Synchronization
- Route Selection

Related Concepts

Transparent Interconnection of Lots of Links (TRILL) mechanisms involve the following concepts:

RBs

RBs are classified into the following types based on their locations and functions in transmitting data packets on a TRILL network:

- Ingress RB: receives data packets from the source host, encapsulates them into TRILL data packets, and sends the TRILL data packets to the TRILL network for further forwarding.
- Transit RB: forwards TRILL data packets.
- Egress RB: decapsulates TRILL data packets into user data packets and sends the data packets to the destination host.

SPF Algorithm

The Shortest Path First (SPF) algorithm calculates the shortest path from a host to each host on a network. After performing the SPF algorithm, a host generates a shortest path tree (SPT) which spreads to all destination hosts.

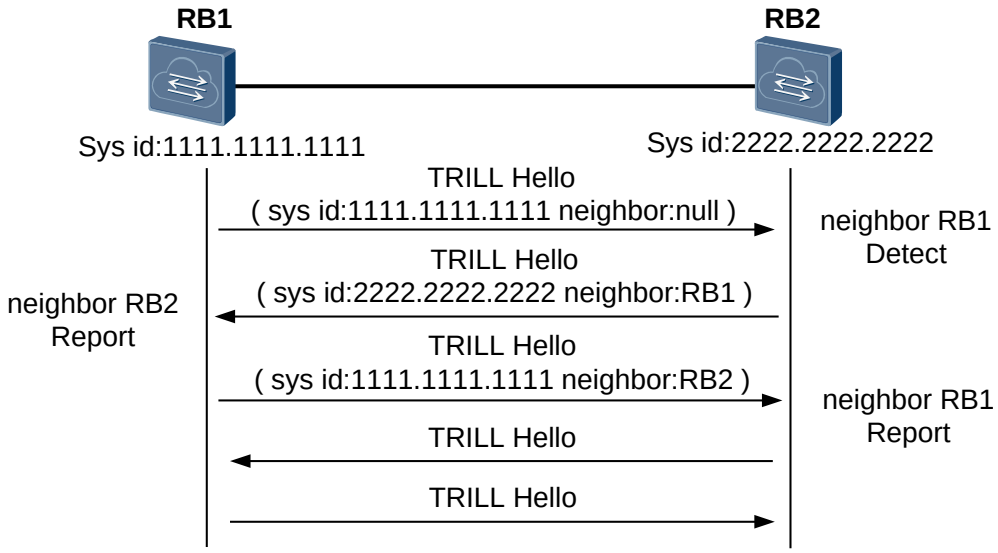
RPF check

After receiving a packet, a device searches the RPF tables for an optimal route to the source IP address of the packet as the Reverse Path Forwarding (RPF) route. If the interface that the packet reaches is the outbound interface of the optimal route, the packet passes the RPF check; otherwise, the packet fails the RPF check and is discarded. For details, see RPF check description in the chapter "Multicast Route Management" in *Enterprise Data Communication Products Feature Description – IP Multicast*.

TRILL Neighbor Relationship Establishment

RBs establish TRILL neighbor relationships by exchanging Hello packets. The structure of Hello packets sent from broadcast interfaces is different from that of packets sent from P2P interfaces. However, the process of establishing TRILL neighbor relationships is similar, regardless of packet structures. Figure 1-5 shows how two RBs establish a TRILL neighbor relationship.

Figure 1-5 Process of establishing a TRILL neighbor relationship



As shown in Figure 1-5, establishing a TRILL neighbor relationship involves the following steps:

1. RB1 sends a TRILL Hello packet to RB2. After receiving the packet, RB2 sets the neighbor status of RB1 to **Detect** if the neighbor field carried in the packet does not contain RB2's MAC address.
2. RB2 adds RB1's MAC address to the neighbor field and replies to RB1 with the TRILL Hello packet. After receiving the packet, RB1 detects its MAC address in the neighbor field and sets the neighbor status of RB2 to **Report**.
3. RB1 adds RB2's MAC address to the neighbor field and sends the TRILL Hello packet to RB2. After receiving the packet, RB2 sets the neighbor status of RB1 to **Report**. By now, the TRILL neighbor relationship is established between RB1 and RB2.
4. After the TRILL neighbor relationship is established, RB1 and RB2 exchange Hello packets periodically to maintain the neighbor relationship. If one end does not receive any response from the other after sending three consecutive Hello packets, this end considers the neighbor Down.

To accelerate route convergence and increase communication efficiency on broadcast networks, TRILL offers the following mechanisms:

- Designated routing bridge (DRB) election

  Each RB on a broadcast network needs to exchange packets with all other RBs. If N RBs exist, then N x (n-1)/2 adjacencies need to be established. If the status of one RB changes, the RB needs to send a large number of packets, wasting bandwidth resources. To address this problem, TRILL introduces the concept of DRB election. The DRB is selected after the neighbor status becomes **Detect**. RBs send packets to the DRB only, and the DRB is responsible for broadcasting the packets.

  A pseudonode is a virtual node on a broadcast network, not an actual RB. A pseudonode is identified by the DRB's system ID and one-byte non-zero circuit ID. Pseudonodes simplify the network topology and shorten the length of LSPs that RBs generate. When the network topology changes, the number of LSPs flooded will be reduced, and so as the SPF resources.

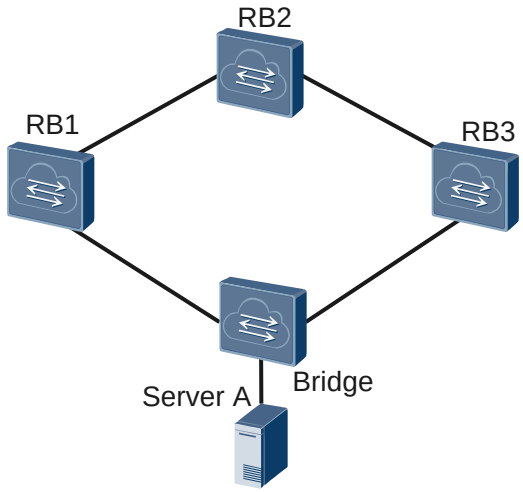
  The DRB is selected based on the following rules in sequence:

1. The interface with the higher DRB priority is elected as the DRB.
2. If both interfaces share the same DRB priority, the interface with the larger MAC address is selected as the DRB.

- Appointed forwarder (AF)

  When multicast packets or unknown unicast packets are forwarded over a TRILL network, loops may occur because these packets can be broadcast within the same VLAN. As shown in Figure 1-6, Host A sends a user multicast packet to the TRILL network through the Layer 2 switch (Bridge). If RB1 and RB3 belong to the same VLAN, the packet will be sent to both RB1 and RB3, causing a loop. The loop can be avoided if an AF is available. The DRB selects an AF based on the CE VLAN, and only the AF can function as an ingress or egress RB. Non-AF RBs can function only as transit RBs. If RB1 in Figure 1-6 is selected as the AF, the packet will be sent to RB1 only, avoiding loops.

Figure 1-6 Networking diagram for selecting an AF



The DRB checks the VLAN to which the user packet belongs and the CE VLANs enabled on the RBs at the ingress of the TRILL network. The RB with the same VLAN as the one to which the user packet belongs is selected as the AF. If more than one RB has the same VLAN as the user packet, the AF is elected based on the following rules in sequence:

1. The RB with the highest DRB priority is elected as the AF.
2. If the RBs share the same DRB priority, the RB with the largest system MAC address is elected as the AF.
3. If the RBs share the same system MAC address, the RB with the largest interface ID is elected as the AF.
4. If the RBs share the same interface ID, the RB with the largest system ID is elected as the AF.

NOTE

- Only the RB with TRILL port mode Access or Hybrid can be elected as the AF.
- If two or more RBs have the same nickname on a broadcast network, none of them can be selected as the AF.
- On a broadcast network, if a DRB changes, the DRB deletes all AF information.

- Designated VLAN (DVLAN)

If there are multiple carrier-VLANs on a TRILL network, a DVLAN must be specified on the interface for traffic forwarding. Before sending protocol packets or forwarding TRILL packets, an interface adds a DVLAN to the VLAN field in the outer Ethernet header. The DVLAN can be configured or specified by the DRB.

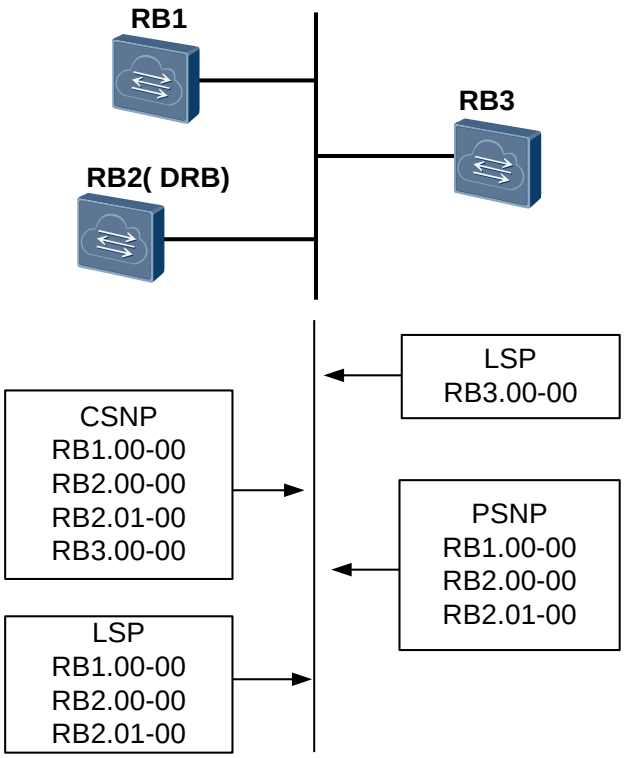
LSDB Synchronization

Link state database (LSDB) synchronization is the process of unifying the LSDB maintained by each RB on a network after the DRB is elected. The LSDB is used to generate the forwarding table. Therefore, LSDBs must be synchronized to ensure that data packets can be transmitted properly. The LSDB synchronization process varies with the network type.

- Figure 1-7 shows the update process on a broadcast link.

1. The newly added RB3 sends Hello packets to establish neighbor relationships with other RBs in the broadcast domain.
2. After neighbor relationships are established, RB3 sends an LSP to multicast address 01-80-C2-00-00-41. All the neighbors on the network will receive this LSP.
3. The DRB on the network segment adds the LSP from RB3 to its LSDB. After the CSNP timer expires, the DRB sends CSNPs to synchronize LSDBs on the network. The default interval at which CSNP packets are transmitted is 10s.
4. After receiving the CSNPs from the DRB, RB3 checks its LSDB and sends a PSNP to request the LSPs that are unavailable in its LSDB.
5. After receiving the PSNP, the DRB sends the required LSPs to synchronize LSDBs. The following describes how the DRB updates its LSDB:
    1. After receiving an LSP that is unavailable in its LSDB, the DRB adds it to its LSDB and broadcasts the updated LSDB.
    2. If the sequence number of the received LSP is greater than that of the corresponding LSP in its LSDB, the DRB replaces the local LSP with the received LSP and broadcasts the new LSDB.
    3. If the sequence number of the received LSP is smaller than that of the corresponding LSP in its LSDB, the DRB sends the local LSP from the inbound interface of the received LSP.
    4. If the sequence number of the received LSP is equal to that of the corresponding LSP in its LSDB, the DRB compares the **Remaining Lifetime** values of the two LSPs. If the **Remaining Lifetime** of the received LSP is smaller than that of the corresponding LSP in its LSDB, the DRB replaces the local LSP with the received LSP and broadcasts the new LSDB. If the **Remaining Lifetime** of the received LSP is greater than that of the corresponding LSP in its LSDB, the DRB sends the local LSP from the inbound interface of the received LSP.
    5. If the received LSP and the corresponding LSP in the LSDB share the same sequence number and **Remaining Lifetime**, the DRB compares the **Checksum** values of the two LSPs. If the **Checksum** of the received LSP is greater than that of the corresponding LSP in its LSDB, the DRB replaces the local LSP with the received LSP and broadcasts the new LSDB. If the **Checksum** of the received LSP is smaller than that of the corresponding LSP in its LSDB, the DRB sends the local LSP from the inbound interface of the received LSP.
    6. If the received LSP and the corresponding LSP in the LSDB share the same sequence number, **Remaining Lifetime**, and **Checksum**, the DRB does not forward the received LSP.

Figure 1-7 Process of updating LSDBs on a broadcast link



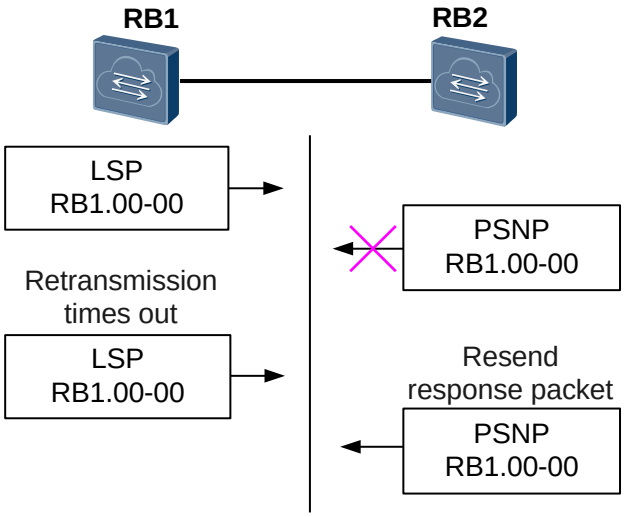
- Figure 1-8 shows the LSDB update process on a P2P link.

1. After a P2P neighbor relationship is established between RB1 and RB2, RB1 sends a CSNP to RB2. If RB2's LSDB is asynchronous with the CSNP, RB2 sends a PSNP to RB1 to request the LSPs that are unavailable in its LSDB.
2. After receiving the PSNP, RB1 replies with the required LSPs and starts the LSP timer at the same time.
3. If RB1 does not receive any PSNP as a response from RB2 before the LSP timer expires, RB1 resends the LSP. After receiving the LSP from RB1, RB2 performs one of the following operations:
    1. If the sequence number of the received LSP is greater than that of the corresponding LSP in its LSDB, RB2 replaces the local LSP with the received LSP, replies to RB1 with a PSNP, and sends the new LSP to other neighbors.
    2. If the sequence number of the received LSP is smaller than that of the corresponding LSP in its LSDB, RB2 sends its LSP to RB1 and waits for a PSNP from RB1.
    3. If the sequence number of the received LSP is equal to that of the corresponding LSP in its LSDB, RB2 compares the **Remaining Lifetime** values of the two LSPs. If the **Remaining Lifetime** of the received LSP is smaller than that of the corresponding LSP in its LSDB, RB2 replaces the local LSP with the received LSP and replies to RB1 with a PSNP. If the **Remaining Lifetime** of the received LSP is greater than that of the corresponding LSP in its LSDB, RB2 replies to RB1 with the local LSP and waits for a PSNP from RB1.
    4. If the received LSP and the corresponding LSP in the LSDB share the same sequence number and **Remaining Lifetime**, RB2 compares the **Checksum** values of the two LSPs. If the **Checksum** of the received LSP is greater than that of the corresponding LSP in its LSDB, RB2 replaces the local LSP with the received LSP

and replies to RB1 with a PSNP. If the **Checksum** of the received LSP is smaller than that of the corresponding LSP in its LSDB, RB2 replies to RB1 with the local LSP and waits for a PSNP from RB1.

5. If the received LSP and the corresponding LSP in the LSDB share the same sequence number, **Remaining Lifetime**, and **Checksum**, RB2 does not forward the received LSP.

Figure 1-8 Process of updating LSDBs on a P2P link



Route Selection

When all the LSDBs on a TRILL network are synchronized, each RB performs the following operations to generate a nickname unicast and multicast routing tables.

- Each RB uses the SPF algorithm to select the shortest path tree (SPT) from itself to each other node, checks neighbor information to identify the outbound interface and next hop for each SPT, and generates a nickname unicast forwarding table.
- In most cases, more than one multicast distribution tree (MDT) is established to transmit multicast services over a TRILL network. The process of generating a nickname multicast routing table is as follows:

1. Root RB selection: Each device identifies the RB with the highest nickname root priority (root RB) and the smallest number (N) of MDTs that can be established on an RB.
2. MDT root selection: The root RB selects MDT roots. If no MDT root is selected, the RBs with top N highest nickname root priorities function as the MDT roots.
3. MDT selection: Each MDT root selects the SPT to each other node from itself.
4. Multicast routing table generation: Each RB checks the MDT information advertised by the ingress RB and generates a multicast routing table for RPF check to avoid loops.
5. Pruning: If no user that joins a specific multicast group accesses an RB, the RB is pruned from the multicast group and traffic of this multicast group is no longer copied to this RB, saving link bandwidth resources.

NOTE

The unicast route to the RB with the highest nickname root priority and each MDT root must be reachable. Therefore, unicast route selection precedes multicast route selection.

1.4.3 TRILL Forwarding Process

On a Transparent Interconnection of Lots of Links (TRILL) network, routing bridges (RBs) exchange Hello packets to establish neighbor relationships and synchronize their link state databases (LSDBs) through link state protocol data unit (LSP) flooding. Each RB checks the LSDB and performs the SPF algorithm to calculate the shortest path tree (SPT) to each node and the outbound interface and next hop and generates a nickname unicast forwarding table.

TRILL transmits user packets based on the MAC address carried in each packet.

- If the MAC address is a unicast address, the TRILL network follows the process of forwarding unicast traffic.
- If the MAC address is a multicast or broadcast address, the TRILL network follows the process of forwarding multicast traffic.

Process of Forwarding Unicast Traffic

Whether unicast traffic carries a known destination address determines its forwarding process. The following list the two forwarding processes:

- Process of forwarding packets with a destination address: Figure 1-9 and Figure 1-10 show how a packet is forwarded from terminal A to C.

Figure 1-9 Process of forwarding known unicast traffic

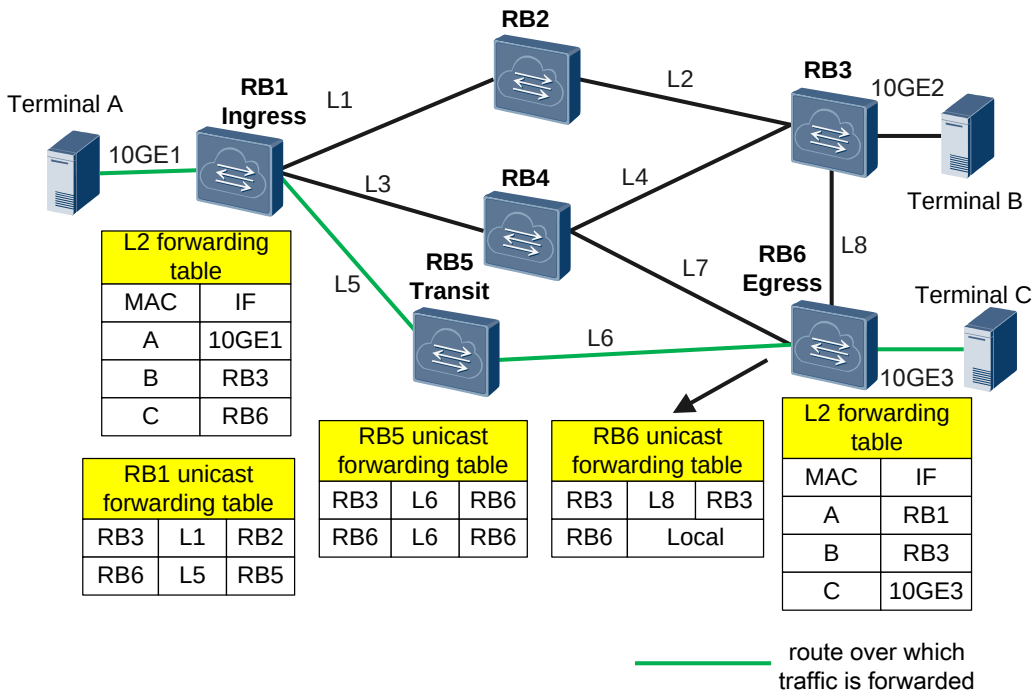
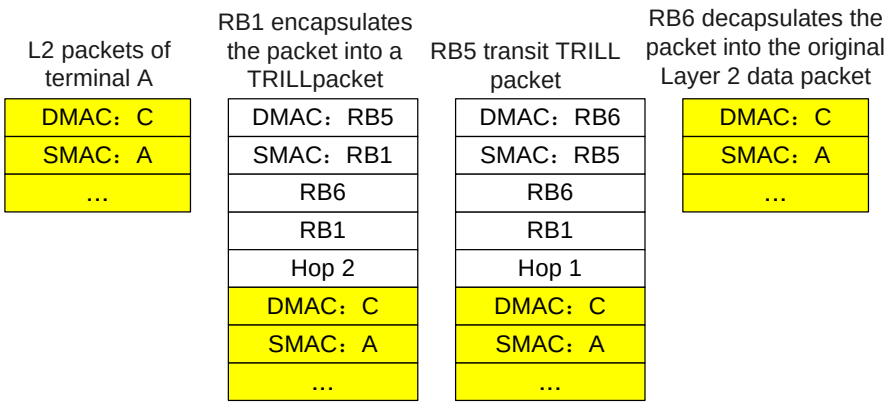


Figure 1-10 Structures of a TRILL unicast packet during transmission



1. After receiving the Layer 2 packet from terminal A, RB1 (ingress RB) searches the Layer 2 forwarding table for the egress nickname name based on the destination MAC address carried in the packet. After that, RB1 searches the unicast forwarding table for the outbound interface (L5) and next hop (RB5), encapsulates the packet into a TRILL data packet, and sends it to RB5.
2. After receiving the TRILL data packet, RB5 (transit RB) parses the packet, searches the unicast forwarding table for the destination based on the egress nickname, and sends the packet to the destination RB (RB6) through the outbound interface (L6).
3. After receiving the TRILL data packet, RB6 (egress RB) parses the packet and identifies the egress (itself). Then RB6 decapsulates the packet into the original Layer 2 data packet and sends it based on the destination MAC address.

- Process of forwarding unknown unicast traffic: Packets are copied to all nodes on the network through multicast distribution.

Process of Forwarding Multicast Traffic

When multicast traffic reaches a TRILL network, the ingress RB selects an MDT to forward the traffic. If more than one next hop exists on the network, the ingress RB copies the traffic and sends the copies to each outbound interface based on the multicast forwarding table. Figure 1-11 and Figure 1-12 shows how a multicast packet is forwarded.

Figure 1-11 Process of forwarding multicast traffic

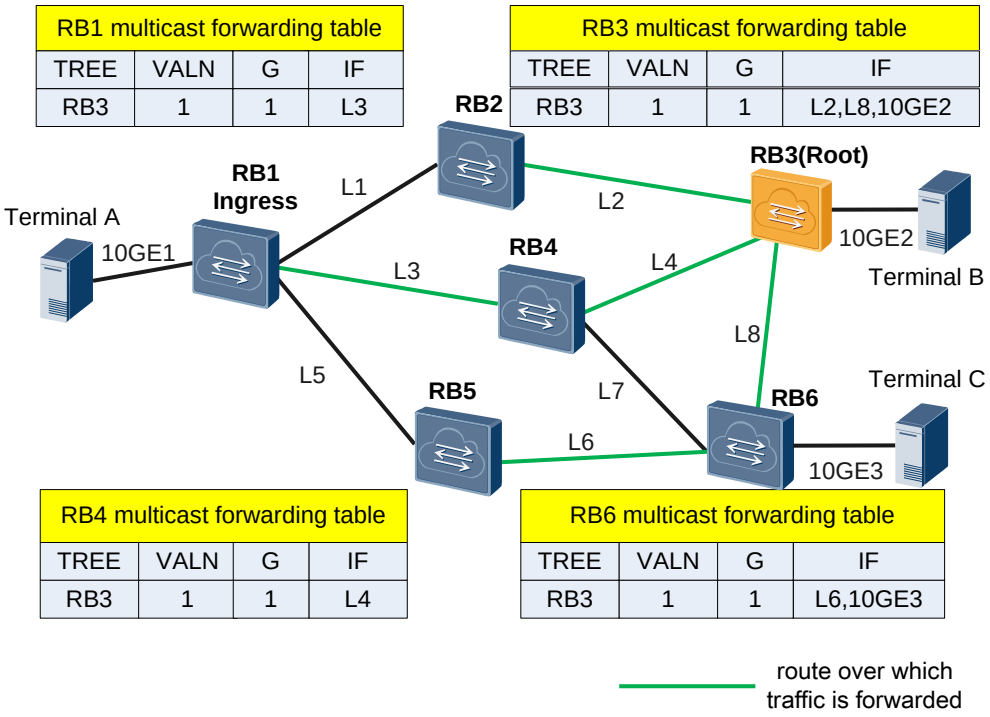
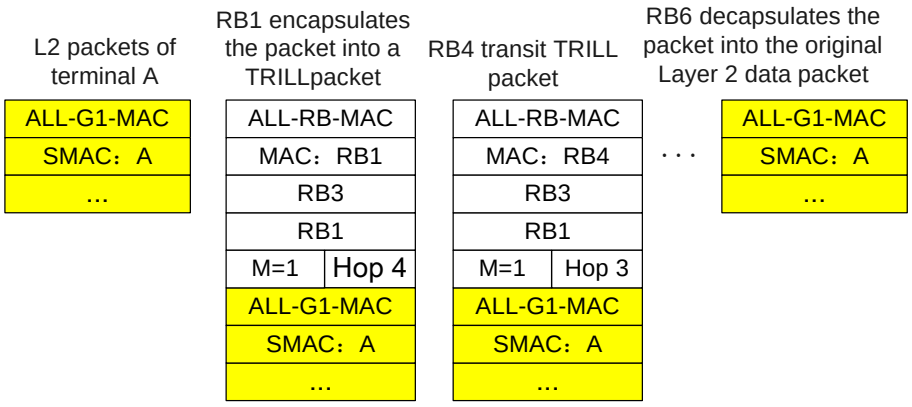


Figure 1-12 Structures of a TRILL multicast packet during transmission



1. After receiving a Layer 2 data packet from terminal A, RB1 (ingress RB) identifies the multicast MAC address carried in the packet and selects an MDT for it based on the VLAN to which the packet belongs. Then RB1 encapsulates the packet into a TRILL data packet and sets the M flag in the TRILL header to 1, indicating that the packet is a multicast packet. After that, RB1 searches the multicast forwarding table based on the nickname of the root RB for an outbound interface to forward the packet.
2. After receiving the TRILL data packet, RB4 (transit RB) parses the TRILL header, identifies the M flag, searches the multicast forwarding table based on the nickname of the egress RB for an outbound interface to forward the packet.
3. After receiving the TRILL data packet, RB3 (root RB) distributes the packet to all outbound interfaces.

4. After receiving the TRILL data packet, RB6 (egress RB) decapsulates the packet into the original Layer 2 data packet and forwards it from the outbound interface.

1.4.4 Load Balancing

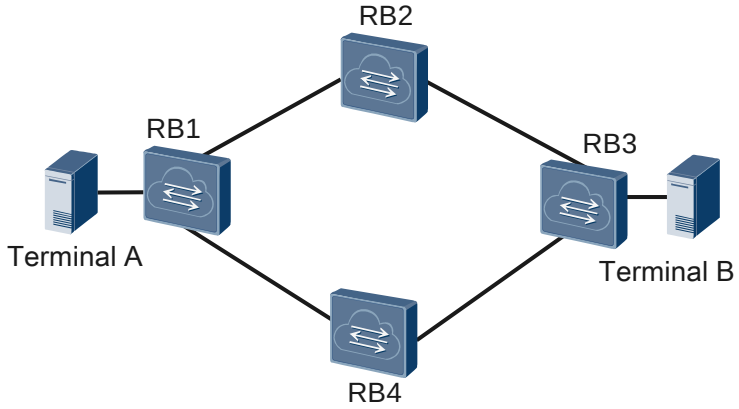
On Transparent Interconnection of Lots of Links (TRILL) networks, multiple equal-cost routes are destined for the same node. If only the optimal route selected using the Shortest Path First (SPF) algorithm transmits traffic, link bandwidth resource usage is relatively low, which cannot meet the requirements for network planning and traffic management. To address this problem, enable equal-cost routes to load-balance the traffic.

The following sections describe the process of load-balancing unicast and multicast traffic.

Process of Load-Balancing Known Unicast Traffic

Equal-cost routes can load-balance unicast traffic on TRILL networks. Before forwarding unicast traffic, a routing bridge (RB) searches the unicast forwarding table for the equal-cost routes and selects a next hop for each packet. As shown in Figure 1-13, two equal-cost routes between RB1 and RB3 load-balance the traffic from terminal A to B.

Figure 1-13 Process of load-balancing known unicast traffic



If equal-cost routes outnumber that supported to load-balance traffic on a TRILL network, the TRILL network selects the equal-cost routes in the following sequence:

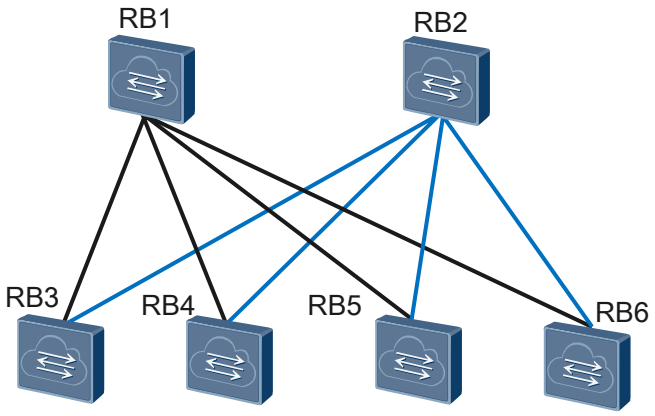
- The route with the smaller outbound interface index is preferred.
- The route of which the next hop RB has the smaller system ID is preferred.

Process of Load-Balancing Multicast Traffic and Unknown Unicast Traffic

Different than the process of load-balancing known unicast traffic, multicast traffic and unknown unicast traffic is transmitted over two multicast distribution trees (MDTs). The two MDTs transmit different broadcast or multicast traffic.

As shown in Figure 1-14, two MDTs are established on the TRILL network, and each RB selects an MDT with the VLANID to transmit traffic.

Figure 1-14 Process of load-balancing multicast traffic and unknown unicast traffic



1.4.5 Dynamic Hostname Mechanism

On Transparent Interconnection of Lots of Links (TRILL) networks, a routing bridge (RB) is identified by its nickname or system ID. In most cases, the nickname (such as 4385) and system ID (such as a023.e304.ff79) are generated by the system automatically. As a result, the nickname and system ID are irregular numbers or a combination of irregular letters and numbers, neither friendly to users nor easy to maintain and manage. To address this problem, you can deploy the dynamic hostname mechanism which provides the mapping between a hostname, the nickname, and system ID.

After a dynamic hostname is configured, the system ID is replaced with the hostname in the following scenarios:

- When TRILL neighbor information is displayed, the system ID of the neighbor is replaced with its hostname even if the neighbor is a designated RB (DRB).
- When an LSP in the TRILL LSDB is displayed, the system ID in the LSP ID is replaced with the hostname of the RB that advertises the LSP.
- When detailed LSDB information is displayed, **Host Name** field is added to the LSP advertised by the RB enabled with dynamic hostname exchange, and the system ID is replaced with the dynamic hostname of the neighbor.

1.4.6 TRILL NSR

As networks develop, value-added services are widely deployed, and network bandwidth increases exponentially. Even a temporary network fault interrupts a large volume of services and brings tremendous loss to data center operators. A master/slave switchover triggered by a system fault or performed by a network administrator when upgrading software or maintaining the system interrupts routing and causes traffic loss. Transparent Interconnection of Lots of Links (TRILL) non-stop routing (NSR) can address this problem.

NSR ensures uninterrupted traffic transmission if a fault occurs on the control plane and a backup control plane is available to take over the traffic. During the switchover, the fault is transparent to the control plane of a neighbor.

Implementation

TRILL NSR uses the following approaches to synchronize data between main control boards in real time:

- TRILL NSR backs up configuration and dynamic data, such as interface, neighbor, and link state database (LSDB) information to the slave control board.
- TRILL NSR does not back up the socket status. The RawLink socket is used to send and receive packets.
- TRILL NSR does not back up data, such as routes and shortest path trees (SPTs). You can use the source data to restore such data during the database backup process.
- When a master/slave switchover occurs, the new master main control board restores the operation data and takes over services from the former master main control board. During the switchover, the fault is transparent to the control plane of a neighbor.

1.4.7 TRILL Authentication

Background

As networks develop, there has been considerable growth in all types of data, voice, and video information exchanged on networks. In addition, new services, such as E-commerce, online conferencing and auctions, video on demand (VoD), and e-learning have sprung up increasingly, requiring higher information security than before. Operators must protect data packets from being intercepted or modified by attackers and prohibit unauthorized users from accessing network resources. Transparent Interconnection of Lots of Links (TRILL) that provides packet encryption and authentication was developed to enhance system security and ensure the proper operating of operator networks.

TRILL supports interface authentication and packets authentication. A local routing bridge (RB) adds an authentication type-length-value (TLV) to a packet before sending it, and the remote RB checks the authentication TLV based on the configured authentication type. The remote RB accepts the packet if it passes the check and discards the packet if it fails the check.

Related Concepts

TRILL Authentication Type

Based on the types of packets to be authenticated, TRILL authentication falls into two types:

- Interface authentication: applies to Hello packets.
- Packets authentication: applies to link state protocol data units (LSPs) and sequence number PDUs (SNPs).

TRILL authentication falls into three modes:

- Simple authentication: The authenticated party adds the configured password directly to packets for authentication. This authentication mode provides the lowest password security among the three modes.
- MD5 authentication: The authenticated party uses the Message Digest 5 (MD5) algorithm to generate a ciphertext password and adds it to packets for authentication. This authentication mode improves password security.
- Keychain authentication: The authenticated party configures a keychain that changes with time. This authentication mode provides the highest password security among the three modes.

Implementation

A TRILL authentication-enabled device adds an authentication field to encrypt a packet before sending it to ensure network security. After receiving a TRILL packet from a remote RB, the local RB discards the packet if the authentication password in the packet is different than the local one. This authentication protects the local RB.

The authentication TLV in a TRILL packet carries authentication information. The meaning of each field in an authentication TLV is described as follows:

- Type: type of a packet to be authenticated. The value in TRILL packets is 133, 1 byte.
- Length: length of an authentication TLV. The value is 1 byte.
- Value: authentication mode (1 byte) and password. The value ranges from 1 to 254 bytes. Authentication mode values are described as follows:
  - 0: reserved
  - 1: simple authentication
  - 54: MD5 authentication
  - 255: private authentication

Interface Authentication

Each interface saves a configured authentication password to authenticate Hello packets in simple, MD5, or keychain mode. Directly connected interfaces must share the same password.

Packets authentication

A local RB adds an authentication TLV to a packet before sending it, and the remote RB checks the authentication TLV based on the configured authentication type. The remote RB accepts the packet if it passes the check and discards the packet if it fails the check.

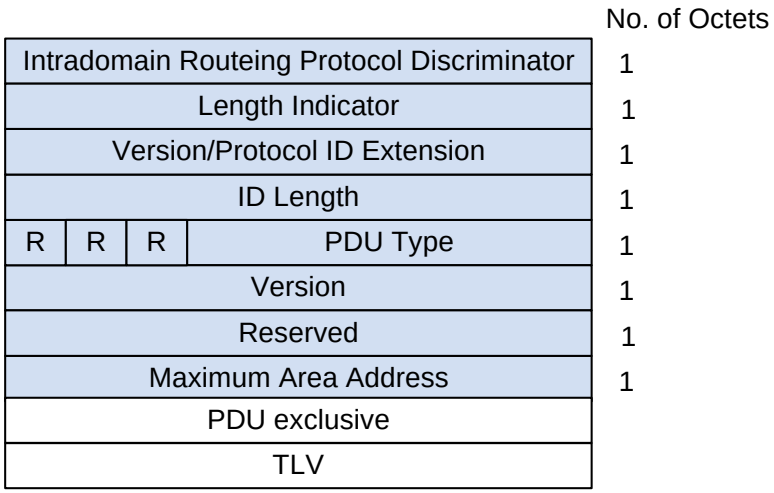
1.4.8 TRILL Control Packets

Switches exchange Transparent Interconnection of Lots of Links (TRILL) control packets to communicate. This section describes TRILL control packets.

TRILL PDU Format

TRILL protocol data units (PDUs) include Hello packets, link state protocol data units (LSPs), and sequence number PDUs (SNPs). The first eight bytes in all TRILL PDUs are the same, as shown in Figure 1-15.

Figure 1-15 TRILL PDU structure



The meanings of main fields are as follows:

- Intradomain Routing Protocol Discriminator: network layer protocol data unit.
- Length Indicator: header length.
- ID Length: length of a system ID.
- PDU Type: type of a PDU.
- Maximum Area Address: the maximum number of area IDs. The TRILL area ID can only be 00.
- TLV: type-length-value. The TLV varies with the PDU type.

Hello Packets

Hello packets are used to establish and maintain neighbor relationships. LAN Hello packets are used on broadcast networks, and P2P Hello packets are used on non-broadcast networks. The two types of Hello packets have different formats.

Figure 1-16 shows the format of a LAN Hello packet on broadcast networks.

Figure 1-16 Format of a LAN Hello packet

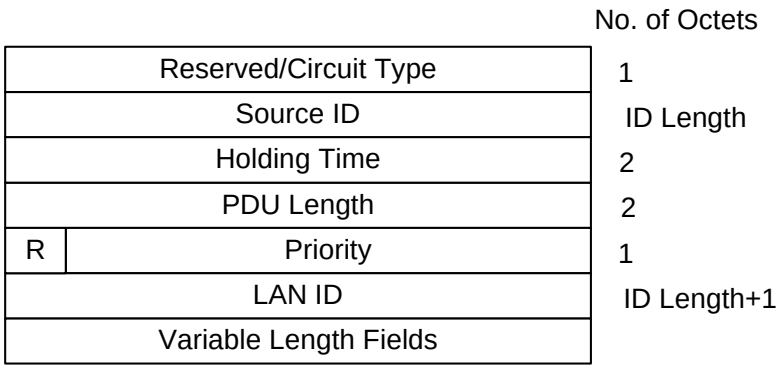


Figure 1-17 shows the format of a P2P Hello packet on non-broadcast networks.

Figure 1-17 Format of a P2P Hello packet

	No. of Octets
Reserved/Circuit Type	1
Source ID	ID Length
Holding Time	2
PDU Length	2
Local Circuit ID	1
Variable Length Fields	

As shown in Figure 1-17, most fields in P2P Hello packets are the same as those in LAN Hello packets. P2P Hello packets do not carry Priority and LAN ID fields but carry a new field, Local Circuit ID.

LSPs

LSPs are used to exchange link state information. Figure 1-18 shows the format of an LSP.

Figure 1-18 Format of a TRILL LSP

				No. of Octets
PDU Length				2
Remaining Lifetime				2
LSP ID				ID Length+2
Sequency Number				4
Checksum				2
R	ATT	OL	IS Type	1
Variable Length Fields				

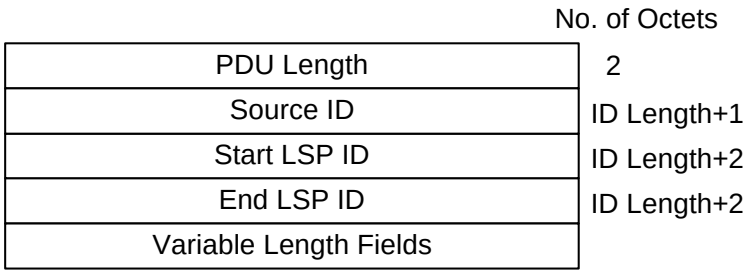
SNPs

SNPs carry complete or partial LSP information and are used to synchronize link state databases (LSDBs). SNPs are classified into two types:

- Complete SNP (CSNP): CSNPs carry summaries of all LSPs in LSDBs to synchronize LSDBs between neighboring switches. On broadcast networks, the DRB sends CSNPs periodically, and the default interval at which CSNPs are sent is 10s. On P2P networks, CSPNs are sent only when two ends are establishing an adjacency for the first time.

  Figure 1-19 shows the format of a CSNP.

Figure 1-19 Format of a TRILL CSNP



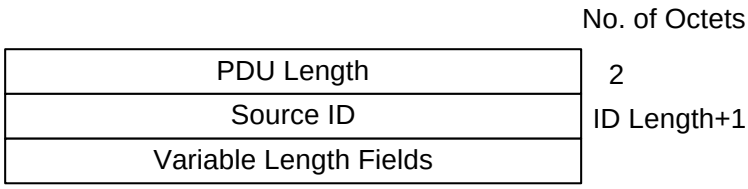
The meanings of main fields are as follows:

- Source ID: system ID of the RB that sends the SNP.
- Start LSP ID: ID of the first LSP in the CSNP.
- End LSP ID: ID of the last LSP in the CSNP.

- Partial SNP (PSNP): PSNPs list only the sequence numbers of recently received LSPs. A PSNP can acknowledge multiple LSPs at a time. If an LSDB is not updated, an RB can use a PSNP to request new LSPs from its neighbor.

  Figure 1-20 shows the format of a PSNP.

Figure 1-20 Format of a TRILL PSNP



1.5 Applications

1.5.1 Application of TRILL in Data Centers

Service Overview

Data centers can speed up communication between Internet infrastructures. Therefore, enterprises and operators are working harder to develop data centers towards the larger scale, virtualization, and cloud computing. In addition, large-scale Layer 2 and virtualization technologies are increasingly used in data centers to offer a large volume of services and reduce the maintenance cost.

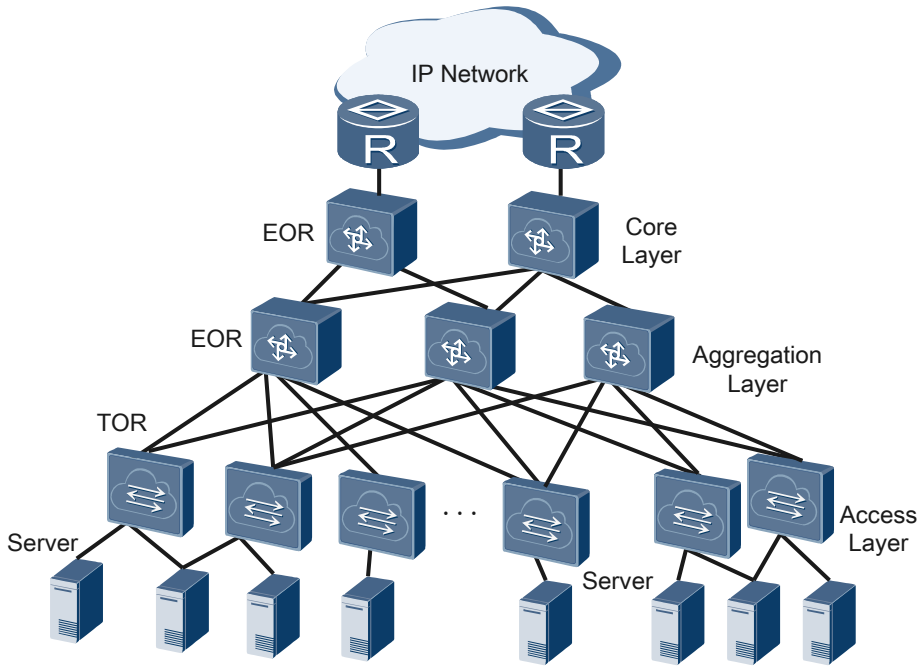
On traditional Layer 2 networks, the relatively low bandwidth usage, low convergence speed, and small network scale are unable to keep pace with the development of data centers, while challenges confront Layer 3 networks in terms of IP address management and virtual device relocation.

As the control protocol on large-scale Layer 2 networks, Transparent Interconnection of Lots of Links (TRILL) combines configuration flexibility of Layer 2 networks and the large scale of Layer 3 networks. It is adaptive to large-scale virtualization of inter-region servers using cloud computing and provides a better solution to data centers.

Networking Description

TRILL data centers are deployed in typical Layer 2 networkings. All access, aggregation, and core switches run TRILL. An access switch can be a Top Of Rack (TOR) or End Of Row (EOR), while aggregation and core switches are EORs in most cases. To ensure that services are processed properly in data centers, operators need to use TRILL to forward traffic efficiently between servers, and between the server and Internet users. Figure 1-21 shows the typical large-scale Layer 2 TRILL networking.

Figure 1-21 Typical data center networking



Feature Deployment

Deploy data center networks using TRILL as follows based on switch roles:

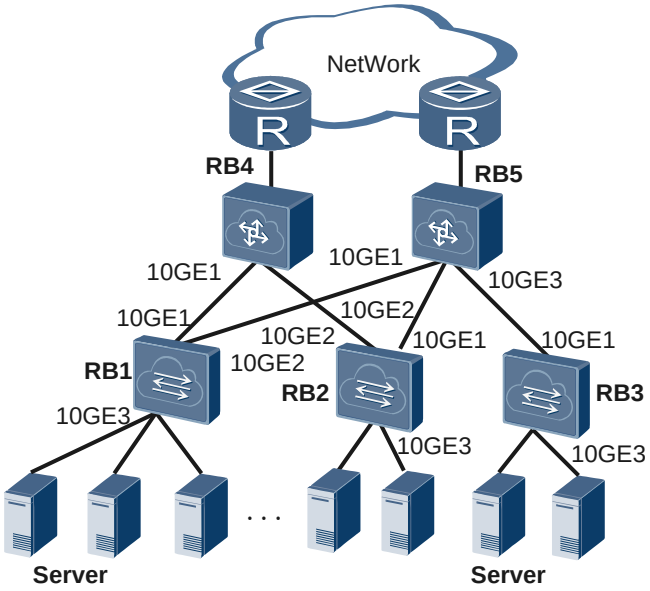
- TRILL user side (access layer): Specify the CE VLAN for TRILL users to ensure security of a TRILL network over which user traffic is transmitted.
- TRILL network side (aggregation and core layers): Enable TRILL on all RBs on TRILL networks and configure Carrier VLANs. Ensure that TRILL networks that need to communicate with each other share the same Carrier VLAN.

TRILL Configuration Example

As shown in Figure 1-22, the five TRILL-capable RBs belong to the same VLAN. A TRILL network is used to implement communication between servers, and between servers and the Layer 3 network. RBs on the TRILL network use the SPF algorithm to generate the unicast

and multicast forwarding tables and transmit traffic based on the forwarding tables. By default, equal-cost routes (if any) load-balance traffic.

Figure 1-22 Networking diagram for basic TRILL configuration



the configuration files of RBs are as follows：

- Configuration file of RB1

```
#
sysname RB1
#
vlan batch 100
#
trill
 network-entity 00.0000.0000.1111.00
 nickname 100
 carrier-vlan 10
 ce-vlan 100
#
interface 10GE1/0/0
 port link-type hybrid
 trill enable
#
interface 10GE2/0/0
 port link-type hybrid
 trill enable
#
interface 10GE3/0/0
 port link-type hybrid
 port hybrid pvid vlan 100
 port hybrid untagged vlan 100
 trill enable port-mode access
#
return
```

- Configuration file of RB2

```
#
sysname RB2
#
vlan batch 100
#
trill
 network-entity 00.0000.0000.2222.00
 nickname 200
 carrier-vlan 10
 ce-vlan 100
#
interface 10GE1/0/0
 port link-type hybrid
 trill enable
#
interface 10GE2/0/0
 port link-type hybrid
 trill enable
#
interface 10GE3/0/0
 port link-type hybrid
 port hybrid pvid vlan 100
 port hybrid untagged vlan 100
 trill enable port-mode access
#
return
```

- Configuration file of RB3

```
#
sysname RB3
#
vlan batch 100
#
trill
 network-entity 00.0000.0000.3333.00
 nickname 300
 carrier-vlan 10
 ce-vlan 100
#
interface 10GE1/0/0
 port link-type hybrid
 trill enable
#
interface 10GE3/0/0
 port link-type hybrid
 port hybrid pvid vlan 100
 port hybrid untagged vlan 100
 trill enable port-mode access
#
return
```

- Configuration file of RB4

```
#
sysname RB4
#
```

```
trill
 network-entity 00.0000.0000.4444.00
 nickname 400
 carrier-vlan 10
#
interface 10GE1/0/0
 trill enable
#
interface 10GE2/0/0
 port link-type hybrid
 trill enable
#
return
```

- Configuration file of RB5

```
#
sysname RB4
#
trill
 network-entity 00.0000.0000.5555.00
 nickname 500
 carrier-vlan 10
#
interface 10GE1/0/0
 port link-type hybrid
 trill enable
#
interface 10GE2/0/0
 port link-type hybrid
 trill enable
#
interface 10GE3/0/0
 port link-type hybrid
 trill enable
#
return
```