Virtual Private Cloud

User Guide

Issue	01
Date	2025-01-24

HUAWEI CLOUD COMPUTING TECHNOLOGIES CO., LTD.

Trademarks and Permissions


and other Huawei trademarks are the property 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 Cloud 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 a warranty of any kind, express or implied.

Contents

1 Permissions Management

1.1 Creating an IAM User and Granting VPC Permissions

This section describes how to use IAM to implement fine-grained permissions control for your VPC resources. With IAM, you can:

- Create IAM users for personnel based on your enterprise's organizational structure. Each IAM user has their own identity credentials for accessing VPC resources.
- Grant users only the permissions required to perform a given task based on their job responsibilities.
- Entrust a HUAWEI ID or cloud service to perform efficient O&M on your VPC resources.

If your HUAWEI ID meets your permissions requirements, you can skip this section.

Figure 1-1 shows the process flow for granting permissions.

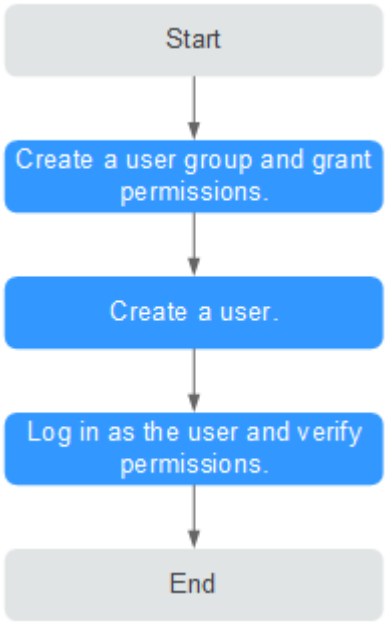
Prerequisites

Learn about the permissions (see **Permissions**) supported by VPC and choose policies or roles according to your requirements.

To grant permissions for other services, learn about all **system-defined permissions** supported by IAM.

Process Flow

Figure 1-1 Process for granting VPC permissions



1. On the IAM console, **create a user group and grant it permissions**.

   Create a user group on the IAM console and assign the **VPCReadOnlyAccess** permissions to the group.
2. **Create an IAM user and add it to the created user group**.

   Create a user on the IAM console and add the user to the group created in **1**.
3. **Log in as the IAM user** and verify permissions.

   In the authorized region, perform the following operations:
   - Choose **Service List** > **Virtual Private Cloud**. Then click **Create VPC** on the VPC console. If a message appears indicating that you have insufficient permissions to perform the operation, the **VPCReadOnlyAccess** policy is in effect.
   - Choose another service from **Service List**. If a message appears indicating that you have insufficient permissions to access the service, the **VPCReadOnlyAccess** policy is in effect.

1.2 VPC Custom Policies

Custom policies can be created to supplement the system-defined policies of VPC. For the actions supported for custom policies, see **Permissions Policies and Supported Actions**.

You can create custom policies in either of the following ways:

- Visual editor: Select cloud services, actions, resources, and request conditions. This does not require knowledge of policy syntax.
- JSON: Edit JSON policies from scratch or based on an existing policy.

For operation details, see **Creating a Custom Policy**. The following section contains examples of common VPC custom policies.

Example Custom Policies

- Example 1: Allowing users to create and view VPCs

```
{
   "Version": "1.1",
   "Statement": [
      {
         "Effect": "Allow",
         "Action": [
            "
                vpc:vpcs:create
                vpc:vpcs:list
             "
         ]
      }
   ]
}
```

- Example 2: Denying VPC deletion

  A deny policy must be used in conjunction with other policies to take effect. If the permissions assigned to a user contain both Allow and Deny actions, the Deny actions take precedence over the Allow actions.

  The following method can be used if you need to assign permissions of the **VPC FullAccess** policy to a user but also forbid the user from deleting VPCs. Create a custom policy for denying VPC deletion, and assign both policies to the group the user belongs to. Then the user can perform all operations on VPC except deleting VPCs. The following is an example deny policy:

```
{
    "Version": "1.1",
    "Statement": [
        {
       "Effect": "Deny",
            "Action": [
                "vpc:vpcs:delete"
            ]
        }
    ]
}
```

- Example 3: Defining permissions for multiple services in a policy

  A custom policy can contain the actions of multiple services that are of the global or project-level type. The following is an example policy containing actions of multiple services:

```
{
   "Version": "1.1",
   "Statement": [
      {
         "Action": [
            "vpc:vpcs:create",
            "vpc:vpcs:update"
         ],
         "Effect": "Allow"
      },
      {
         "Action": [
            "ecs:servers:delete"
         ],
         "Effect": "Allow"
      }
   ]
}
```


- Example 4: Allowing users to view associated resources

  To allow users to view resources associated with a specific resource, you need to assign them permissions to query that resource and its associated resources. The following is an example policy containing actions for allowing users to view the servers, extended network interfaces, and supplementary network interfaces associated with a security group:

```
{
   "Version": "1.1",
   "Statement": [
      {
         "Effect": "Allow",
         "Action": [
                "vpc:ports:get",
                "vpc:securityGroups:get",
                "vpc:subNetworkInterfaces:list"
             "
         ]
      }
   ]
}
```


2 VPC and Subnet

2.1 VPC and Subnet Planning

Before using VPCs and subnets to build cloud networks, determine how many VPCs and subnets do you need and plan the necessary CIDR blocks and connectivity options. If you need to connect different VPCs or connect a VPC to an on-premises data center, ensure that their CIDR blocks do not conflict. Properly plan your VPCs and subnets based on the guidelines provided here to avoid CIDR block conflicts, which will make future network expansion easier.

- **How Do I Determine How Many VPCs I Need?**
- **How Do I Determine How Many Subnets I Need?**
- **How Do I Plan CIDR Blocks for VPCs and Subnets?**
- **How Do I Plan How Many Route Tables I Need?**
- **How Do I Connect Two VPC or Connect a VPC to an On-premises Data Center?**

How Do I Determine How Many VPCs I Need?

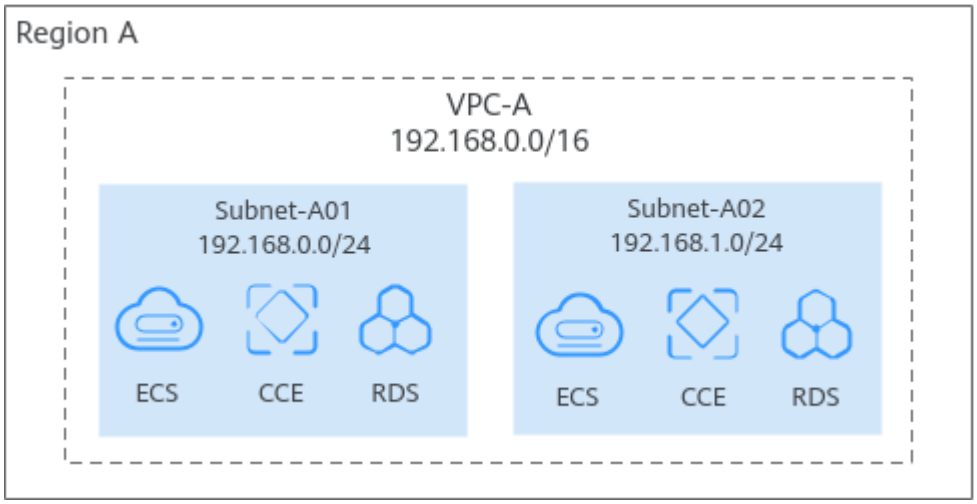
VPCs are region-specific. Cloud resources, such as ECSs, CCEs, and RDS instances, in a VPC must be in the same region as the VPC. By default, different VPCs are isolated from each other, but the subnets in a VPC can communicate with each other.

Planning a Single VPC

If your services are deployed in one region and do not have to handle a lot of traffic, you may not need network isolation. In this case, a single VPC should be enough.

You can create multiple subnets in a VPC for workloads with different requirements and associate route tables with these subnets to control traffic in and out of the subnets. In **Figure 2-1**, services are deployed on different subnets in a VPC (VPC-A in this example).

Figure 2-1 Planning a single VPC



Planning Multiple VPCs

You need to plan multiple VPCs if you have:

- **Services that need to be deployed in different regions**

  VPC is a region-specific service, so services cannot be deployed across regions in a VPC. If your services are deployed in multiple regions, plan at least one VPC in each region.

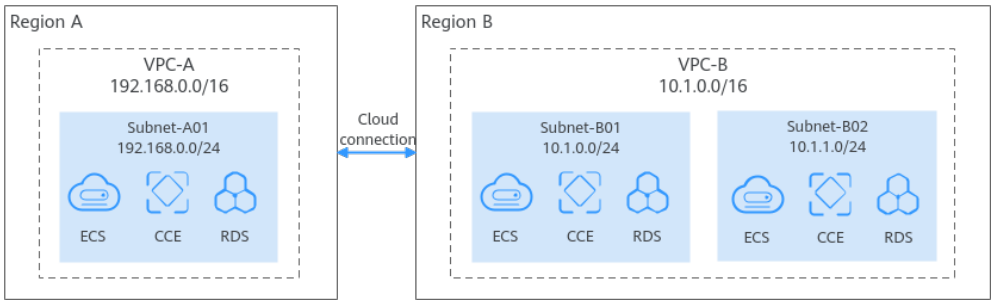
  VPCs are isolated from each other. You can use:

  – A **VPC peering connection** or an **enterprise router** to connect different VPCs in the same region.

  – A **cloud connection** to connect VPCs in different regions.

  In **Figure 2-2**, some services are deployed in VPC-A in region A, and some are deployed in VPC-B in region B. A cloud connection is used to connect VPC-A and VPC-B.

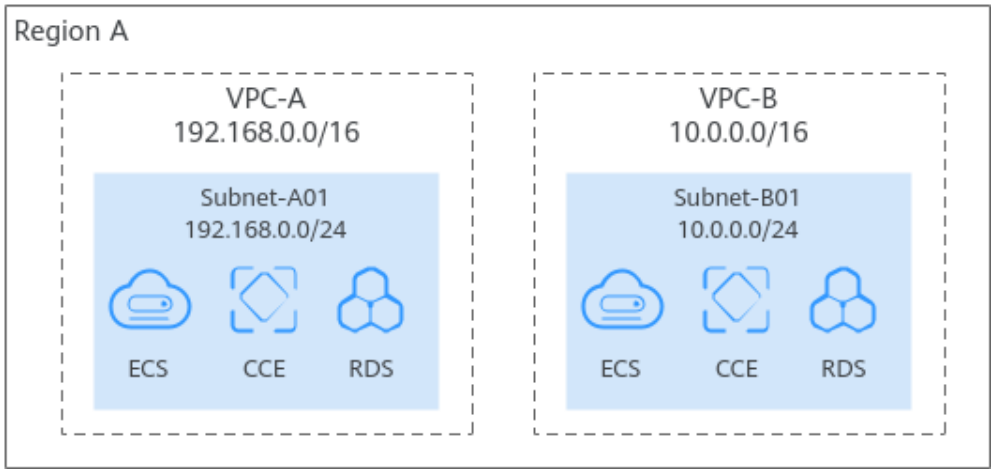
  Figure 2-2 Planning multiple VPCs

- **Services that are deployed in the same region but need network isolation.**

  If your services are deployed in the same region but need network isolation, you need to plan multiple VPCs in this region. Different VPCs are isolated from each other, so you can deploy different services in different VPCs, as shown in **Figure 2-3**. In the figure, some services are deployed in VPC-A, and some are deployed in VPC-B. The two VPCs are isolated from each other.

Figure 2-3 Planning multiple VPCs



NOTE

By default, you can create a maximum of five VPCs in each region. If this cannot meet your service requirements, **request a quota increase**.

How Do I Determine How Many Subnets I Need?

A subnet is a unique CIDR block with a range of IP addresses in a VPC. All cloud resources in a VPC must be deployed on subnets.

You can create different subnets for different services in a VPC. For example, you can create three subnets in a VPC, one subnet for web services, one for management services, and the third one for data services. Additionally, you can use network ACLs to control access to each subnet.

Note the following when selecting subnets and AZs for your resources:

- All instances in different subnets of the same VPC can communicate with each other by default, and the subnets can be located in different AZs. If you have a VPC with two subnets in it and they are located in different AZs, they can communicate with each other by default.
- A cloud resource can be in a different AZ from its subnet. For example, a cloud server in AZ 1 can be in a subnet in AZ 3. If AZ 3 becomes faulty, cloud servers in AZ 1 can still use the subnet in AZ 3, and your services are not interrupted.

NOTE

By default, you can create a maximum of 100 subnets in each region. If this cannot meet your service requirements, **request a quota increase**.

How Do I Plan CIDR Blocks for VPCs and Subnets?

After VPCs and subnets are created, their CIDR blocks cannot be changed. To ensure smooth service expansion and O&M, properly plan VPC and subnet CIDR blocks that best suit your service size and communication requirements.

NOTE

Both IPv4 and IPv6 CIDR blocks can be assigned to a subnet. You can customize IPv4 CIDR blocks but not IPv6 CIDR blocks. The system assigns an IPv6 CIDR block with a 64-bit mask to each subnet, for example, 2407:c080:802:1b32::/64.

Planning VPC CIDR Blocks

When creating a VPC, you need to specify an IPv4 CIDR block for it. Consider the following when selecting a CIDR block:

- Reserve sufficient IP addresses for subsequent service expansion.
- Avoid CIDR block conflicts. To enable communications between VPCs or between a VPC and an on-premises data center, ensure their CIDR blocks do not overlap.

When you create a VPC, you specify a primary IPv4 CIDR block for the VPC, which cannot be changed. You can **add a secondary IPv4 CIDR block to the VPC** if required.

When you create a VPC, we recommend that you use the private IPv4 address ranges specified in **RFC 1918** as the CIDR block, as described in **Table 2-1**.

Table 2-1 VPC CIDR blocks (RFC 1918)

VPC CIDR Block	IP Address Range	Netmask	Example CIDR Block
10.0.0.0/8-24	10.0.0.0–10.255.255.255	8-24	10.0.0.0/8
172.16.0.0/12-24	172.16.0.0–172.31.255.255	12-24	172.30.0.0/16
192.168.0.0/16-24	192.168.0.0–192.168.255.255	16-24	192.168.0.0/24

In addition to the preceding addresses, you can create a VPC with a publicly routable CIDR block that falls outside of the private IPv4 address ranges specified in RFC 1918. However, the reserved system and public CIDR blocks listed in **Table 2-2** must be excluded:

Table 2-2 Reserved system and public CIDR blocks

Reserved System CIDR Blocks	Reserved Public CIDR Blocks
• 100.64.0.0/10 • 214.0.0.0/7 • 198.18.0.0/15 • 169.254.0.0/16	• 0.0.0.0/8 • 127.0.0.0/8 • 240.0.0.0/4 • 255.255.255.255/32

Planning Subnet CIDR Blocks

- Subnet mask planning: The subnet CIDR block must be within the VPC CIDR block. Subnet CIDR blocks in a VPC must be unique. A subnet mask can be between the netmask of its VPC CIDR block and a /29 netmask. If a VPC CIDR block is 10.0.0.0/16, its subnet mask can be anything from 16 to 29.

  For example, if the CIDR block of a VPC is 10.0.0.0/16, you can specify 10.0.0.0/24 for a subnet in this VPC, 10.0.1.0/24 for the second subnet, and 10.0.2.0/24 for the third subnet.
- Planning CIDR block size: After a subnet is created, the CIDR block cannot be changed. You need to properly plan the CIDR block in advance based on the number of IP addresses required by your service.
  - The subnet CIDR block size cannot be too small. Ensure that the number of available IP addresses in the subnet meets service requirements. Remember that the first and last three addresses in a subnet CIDR block are reserved for system use. For example, in subnet 10.0.0.0/24, 10.0.0.1 is the gateway address, 10.0.0.253 is the system interface address, 10.0.0.254 is used by DHCP, and 10.0.0.255 is the broadcast address.
  - The subnet CIDR block cannot be too large, either. If you use a CIDR block that is too large, you may not have enough CIDR blocks available later for new subnets, which can be a problem when you want to scale out services.
- Avoiding subnet CIDR block conflicts: Avoid CIDR block conflicts if you need to connect two VPCs or connect a VPC to an on-premises data center.

  If the subnet CIDR blocks at both ends of the network conflict, **create a subnet**.

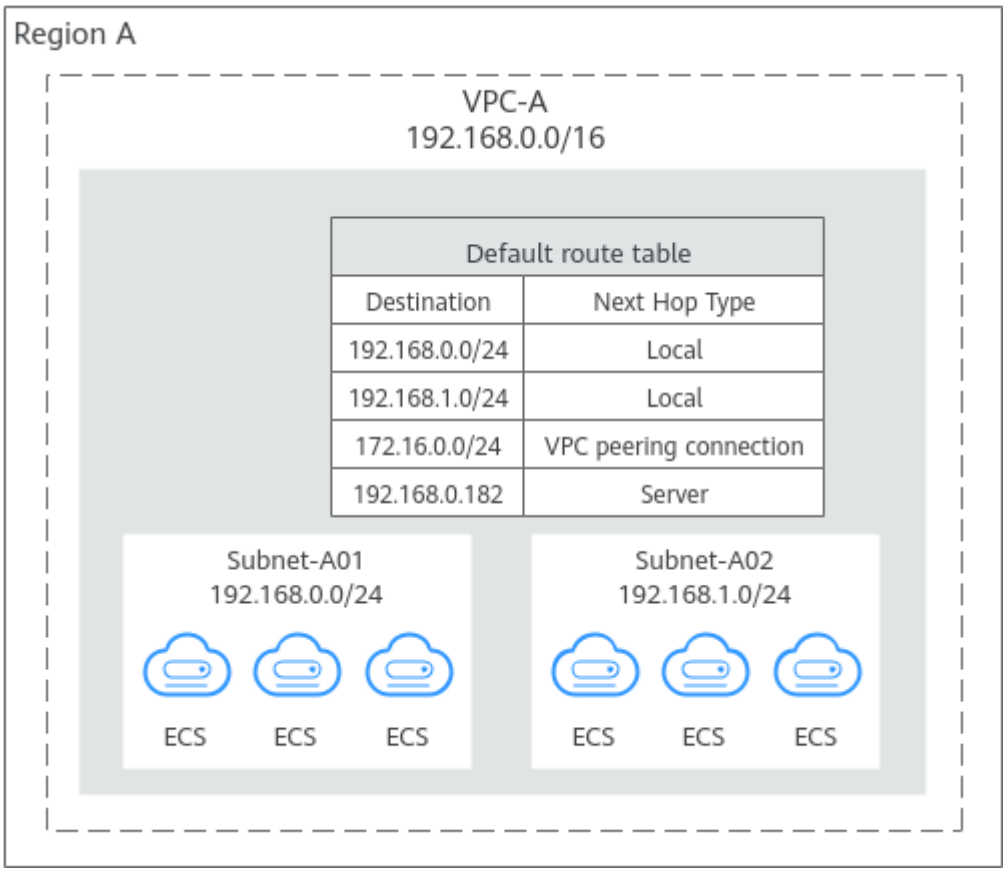
How Do I Plan How Many Route Tables I Need?

A route table contains a set of routes that are used to control the traffic in and out of your subnets in a VPC. You can configure destination, next hop, and other information for each route. A VPC can have multiple route tables. Plan route tables based on the following sections.

Planning One Route Table

If you have the same or similar requirements for controlling the network traffic to and from subnets in a VPC, you can create one route table and associate it with these subnets in this VPC. Each VPC comes with a default route table. If you create a subnet in the VPC, the subnet is associated with the default route table. You can add routes to the default route table to control where the traffic is directed. In **Figure 2-4**, VPC-A has only the default route table, and subnets Subnet-A01 and Subnet-A02 are associated with the default route table.

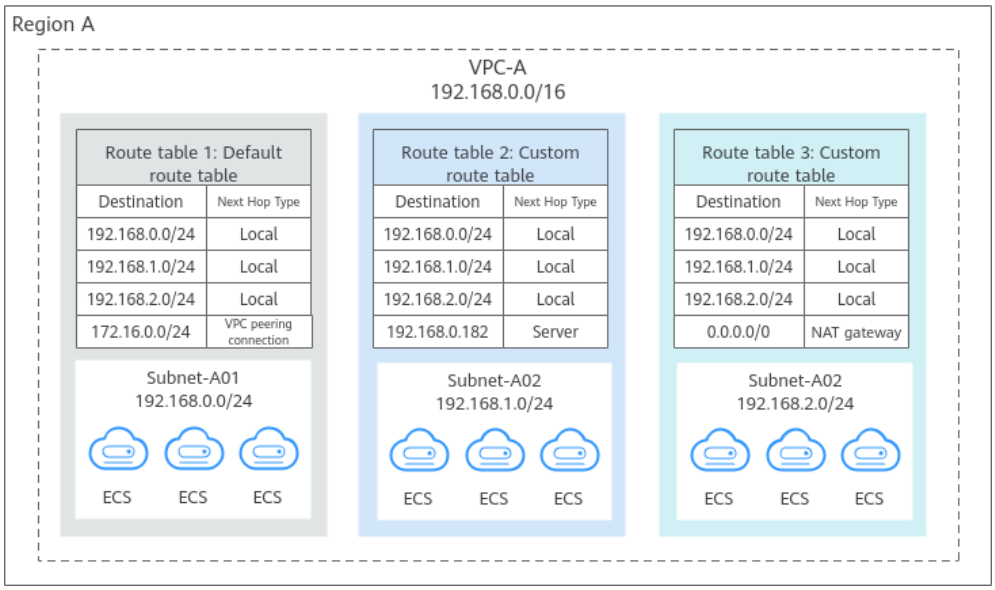
Figure 2-4 Planning one route table



Planning Multiple Route Tables

If you have different requirements for controlling the network traffic to and from subnets in a VPC, the default route table is not enough. You can create one or more custom route tables and associate them with these subnets in this VPC. In **Figure 2-5**, VPC-A has three route tables. Subnet-A01 is associated with default route table 1, Subnet-A02 is associated with custom route table 2, and Subnet-A03 is associated with custom route table 3.

Figure 2-5 Planning multiple route tables



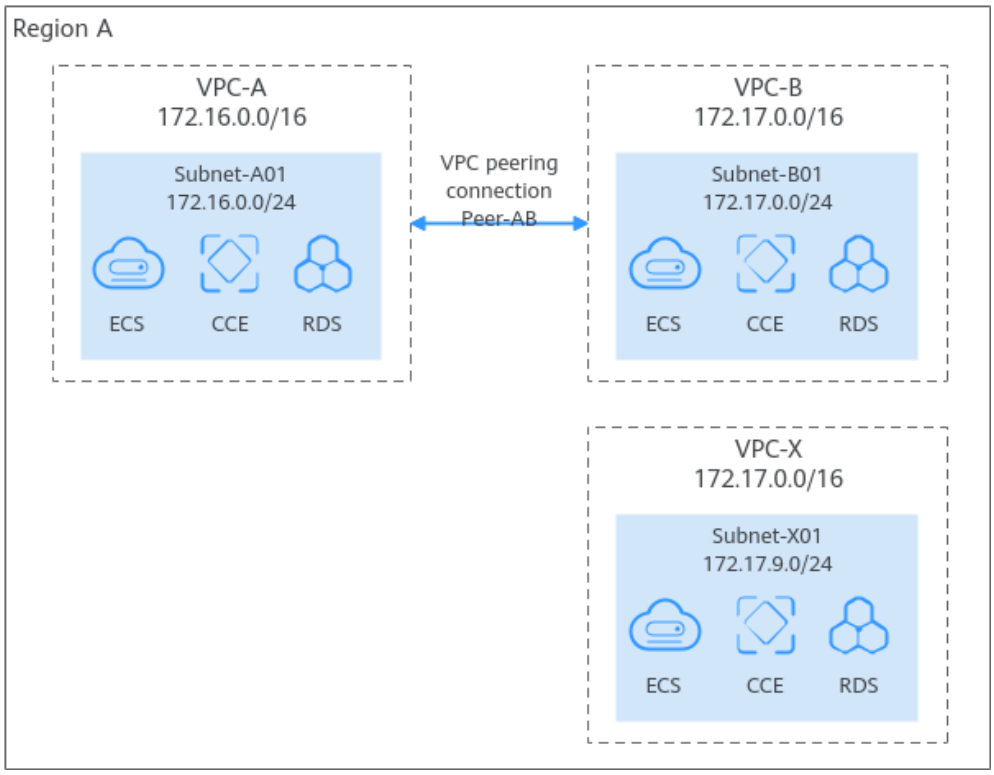
How Do I Connect Two VPC or Connect a VPC to an On-premises Data Center?

If you need to connect two VPCs or connect a VPC to an on-premises data center, ensure that their VPC CIDR blocks do not conflict.

Connecting Two VPCs

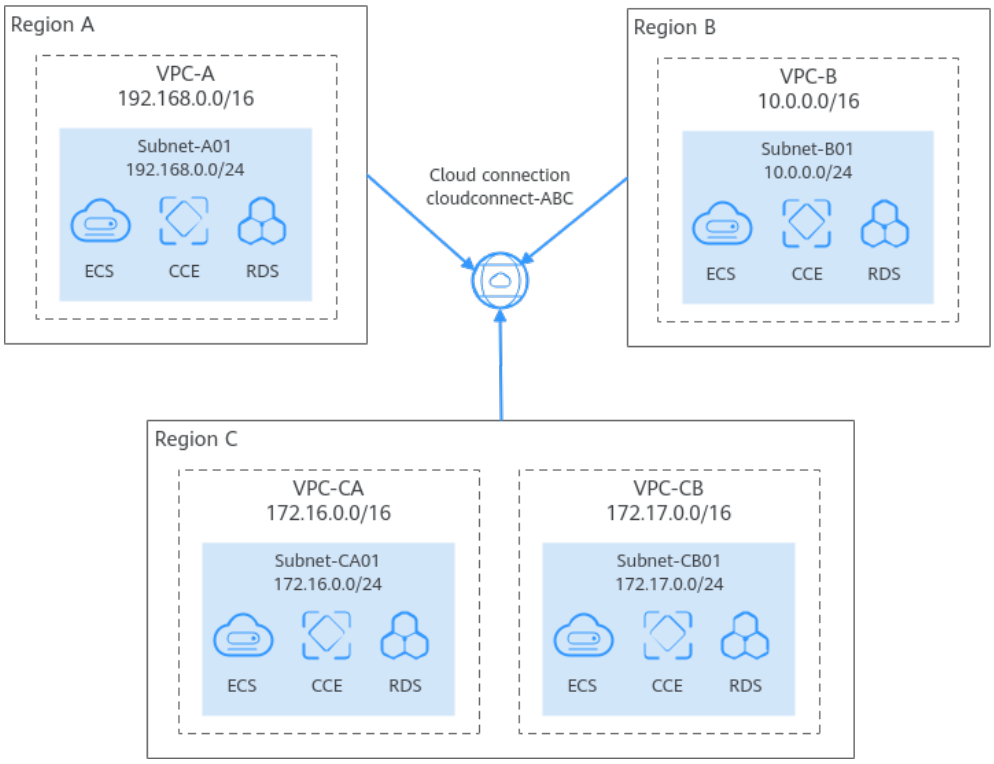
- Connecting VPCs in the same region: In **Figure 2-6**, there are three VPCs in region A: VPC-A, VPC-B, and VPC-X. If you want to connect VPC-A and VPC-B, but isolate VPC-C from other VPCs:
  - Ensure that the CIDR blocks of VPC-A and VPC-B connected by a peering connection (Peering-AB in this example) must be unique.
  - You do not need to worry about VPC CIDR block conflicts because VPC-X does not need to communicate with other VPCs. If VPC-X and VPC-B need to communicate with each other, you can specify different CIDR blocks for the subnets in the two VPCs and create a VPC peering connection to connect the subnets.

Figure 2-6 Connecting VPCs in the same region



- Connecting VPCs in different regions: In **Figure 2-7**, services are deployed in four VPCs in different regions: VPC-A, VPC-B, VPC-CA, and VPC-CB. You can use a cloud connection to connect these VPCs in different regions and ensure that the CIDR blocks of the four VPCs do not conflict.

Figure 2-7 Connecting VPCs in different regions

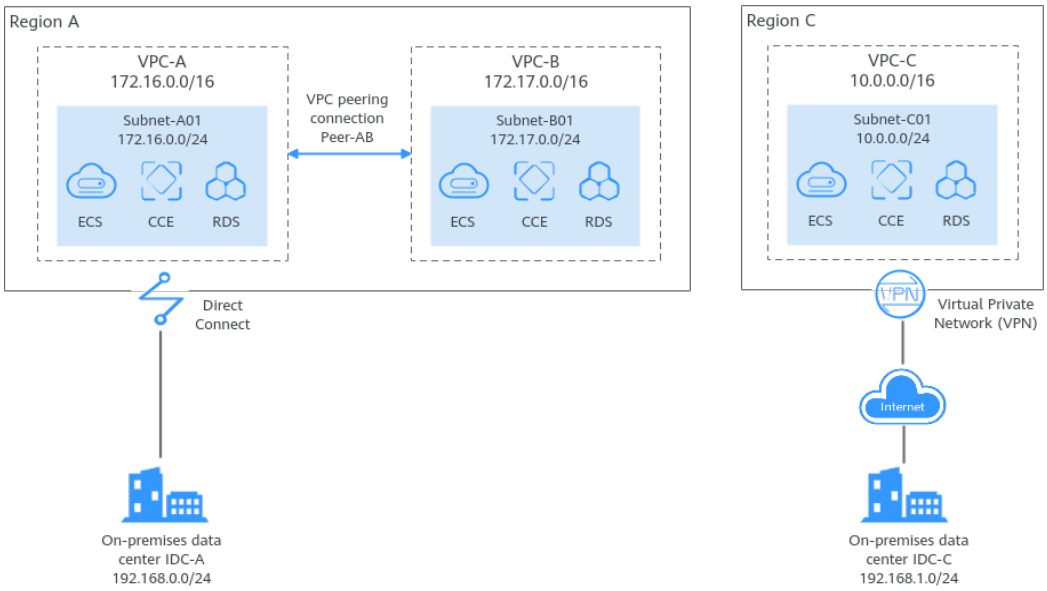


Connecting a VPC to an On-premises Data Center

In **Figure 2-8**, VPC-A and VPC-B in region A need to communicate with each other, and VPC-A needs to connect to on-premises data center IDC-A. In region C, VPC-C needs to connect to on-premises data center IDC-C.

- In region A, VPC-A and VPC-B have different CIDR blocks and can communicate with each other through a VPC peering connection. VPC-A and IDC-A have different CIDR blocks and are connected through a direct connection.
- In region C, VPC-C and IDC-C have different CIDR blocks and are connected through a VPC connection.

Figure 2-8 Connecting a VPC to an on-premises data center



Helpful Link

- You can create a VPC and an ECS to set up an IPv4 private network on the cloud and then bind an EIP to the ECS to allow the ECS to access the Internet. For details, see **Setting Up an IPv4 Network in a VPC**.
- You can create a VPC with an IPv4 and IPv6 CIDR block and create an ECS with both IPv4 and IPv6 addresses in the VPC. You can bind an EIP and add the IPv6 address of the ECS to a shared bandwidth to enable the ECS to communicate with the Internet over both IPv4 and IPv6 networks. For details, see **Setting Up an IPv4/IPv6 Dual-Stack Network in a VPC**.

2.2 VPC Connectivity Options

2.2.1 Overview

Huawei Cloud provides various network services for you to set up secure and scalable cloud networks. With these network services, you can connect VPCs in the same region or different regions, enable the instances (such as ECSs and RDS instances) in VPCs to access the public network, and enable on-premises data centers to access the VPCs. The following describes the function and highlights of each network service. You can flexibly configure VPC and other network services based on your network requirements:

- **Connecting VPCs**
- **Connecting VPCs to the Public Network**
- **Connecting VPCs to an On-Premises Data Center**

Connecting VPCs

With the networking services described in **Table 2-3**, you can flexibly connect VPCs in the same region, in different regions, or in different accounts.

Table 2-3 Networking services that can connect VPCs

Networking Service	Function	Highlights
VPC Peering	With VPC Peering, you can peer two VPCs in the same region. The VPCs can be in the same account or different accounts.	• VPC Peering is free. • Routes can be configured on the console easily.
Enterprise Router	An enterprise router can connect multiple VPCs in the same account or different accounts to set up a hub-and-spoke network. Compared with VPC Peering, Enterprise Router is more suitable for complex networking where many VPCs need to be connected.	• VPCs in the same region can be connected in minutes. • Routes can be automatically added. • Low latency and high speed • Simple network topology and high scalability
Cloud Connect • **Cloud Connection** • **Central Network**	Cloud Connect can connect VPCs in the same account or different accounts across regions. Cloud Connect provides two options: • Cloud connection: Load VPCs in different regions to a cloud connection. • Central network: Attach VPCs in the same region to an enterprise router, and add enterprise routers in different regions to a central network as attachments. This solution features higher scalability and is suitable for complex networking with many VPCs from different regions.	• VPC in different regions can be connected in minutes. • Routes can be automatically added. • Low latency and high speed
VPN	You can use **VPN** connect VPCs in different regions, so that they can communicate with each other over the Internet.	• Low costs • Simple configuration • Immediate use • Unstable networks dependent on the Internet quality

Networking Service	Function	Highlights
Direct Connect	You can use Direct Connect to connect VPCs in different regions.	• Dedicated connections with high security • Low latency and high speed

Connecting VPCs to the Public Network

With the network services described in Table 2-4, you can connect VPCs to the public network so that instances in the VPCs can access the public network or provide services accessible on the public network.

Table 2-4 Network services that allow VPCs to communicate with the public network

Network Service	Function	Highlights
EIP	An EIP is an independent public IP address. You can bind it to an instance, such as an ECS, a NAT gateway, or a load balancer, so that the instance can access the public network or provide services accessible from the public network.	• EIPs can be bound to or unbound from instances if needed. • Shared bandwidths and shared data packages can be used to lower costs. • EIP bandwidth can be adjusted at any time.

Network Service	Function	Highlights
NAT Gateway • **SNAT** • **DNAT**	NAT Gateway supports both source NAT (SNAT) and destination NAT (DNAT). • SNAT enables multiple instances to share one or more EIPs to access the public network. – **ECSs in the same VPC sharing an EIP** – **ECSs in different VPCs sharing an EIP** • DNAT enables port forwarding. It maps EIP ports to ECS ports so that the ECSs in a VPC can share the same EIP and bandwidth to provide Internet-accessible services. However, DNAT does not balance traffic.	• Using shared EIPs to access the public network reduces the costs. • EIPs of ECSs are not exposed to the public network, which improves security. • Different specifications are available.
ELB	**ELB** evenly distributes incoming traffic to multiple backend servers. Together with EIPs, ELB allows a large number of users to access services deployed on cloud servers from the public network.	• ELB can process both Layer 4 and Layer 7 requests and supports advanced forwarding policies and multiple protocols. • ELB can eliminate single points of failure (SPOFs) for high availability.

Connecting VPCs to an On-Premises Data Center

If you have an on-premises data center and not all your workloads can be migrated to the cloud, you can use the network services described in **Table 2-5** to connect your on-premises data center to the VPCs.

Table 2-5 Networking services that can connect VPCs to an on-premises data center

Networking Service	Function	Highlights
VPN	VPN provides an encrypted, Internet-based channel that connects an on-premises data center and the cloud.	• Low costs • Simple configuration • Immediate use • The network quality depends on the Internet.
Direct Connect	Direct Connect establishes a dedicated network connection between an on-premises data center and the cloud.	• Dedicated connections with high security • Low latency and high speed
VPC Peering	With VPC Peering, you can peer two VPCs in the same region, no matter whether they are in the same account or different accounts. VPC Peering can work with Direct Connect or VPN to enable your on-premises data center to access multiple VPCs.	• VPC Peering is free. • Routes can be configured on the console easily.
Enterprise Router	You can use VPN or Direct Connect to connect an on-premises data center to a VPC, and then use an enterprise router to connect multiple VPCs if there are in the same region.	• Route learning is supported. There is no need to configure routes manually. • Multiple connections work in load balancing or active/standby mode for higher availability.

Networking Service	Function	Highlights
Cloud Connect • Cloud Connection • Central Network	You can use Direct Connect or VPN to connect on-premises data centers to VPCs in multiple regions and use a cloud connection or central network to connect the VPCs, so that the on-premises data centers can access all the VPCs. By working with Direct Connect, Cloud Connect provides the following two options: • Cloud connection: Load VPCs in different regions to a cloud connection. For this to work, you need to create a virtual gateway for each VPC that needs to communicate with the on-premises data centers. • Central network: Attach VPCs and Direct Connect global DC gateways in the same region to an enterprise router, and then add the enterprise routers in different regions to a central network. In this way, VPCs in different regions can communicate with the on-premises data centers in multiple cities. Compared with a cloud connection, a central network features a simpler network architecture and higher scalability.	• Route learning is supported. There is no need to configure routes manually. • Network connection policies can be defined flexibly.

2.2.2 Connecting VPCs

Connecting VPCs in the Same Region

If the VPCs you want to connect are in the same region, you can use VPC Peering or Enterprise Router.

Connecting VPCs provides details about different network services.

NOTICE

Before connecting VPCs, you need to plan their CIDR blocks in advance. Overlapping CIDR blocks may cause communication failure.

VPC Peering

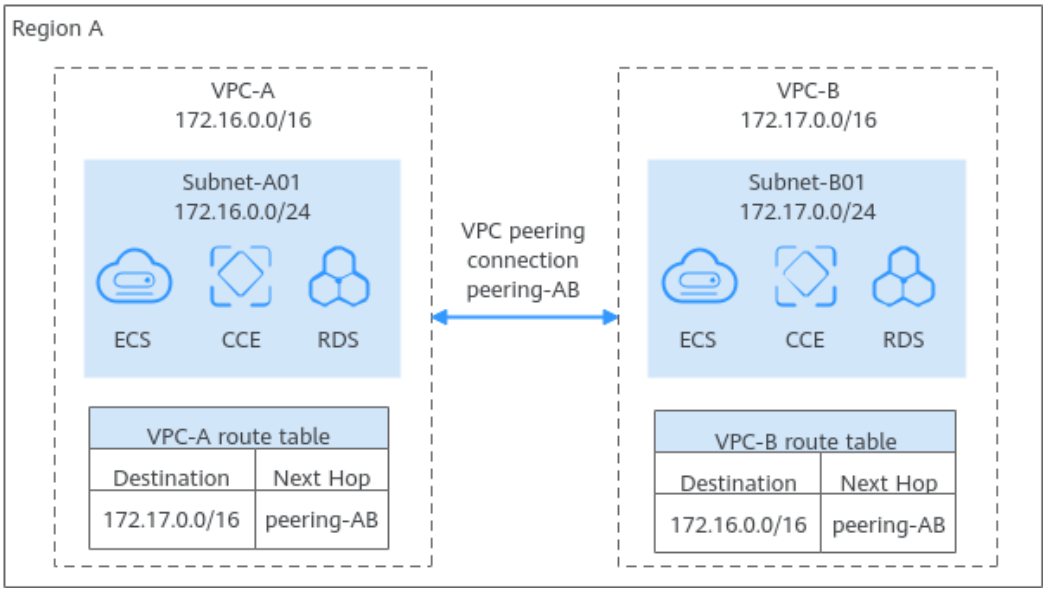
With VPC Peering, you can peer two VPCs in the same region. The VPCs can be in the same account or different accounts.

You can refer to the following topics:

- **Using a VPC Peering Connection to Connect Two VPCs in the Same Account**
- **Using a VPC Peering Connection to Connect Two VPCs in Different Accounts**

In **Figure 2-9**, a VPC peering connection (Peering-AB) connects two VPCs (VPC-A and VPC-B) in a region.

Figure 2-9 Connecting VPCs in the same region over a VPC peering connection



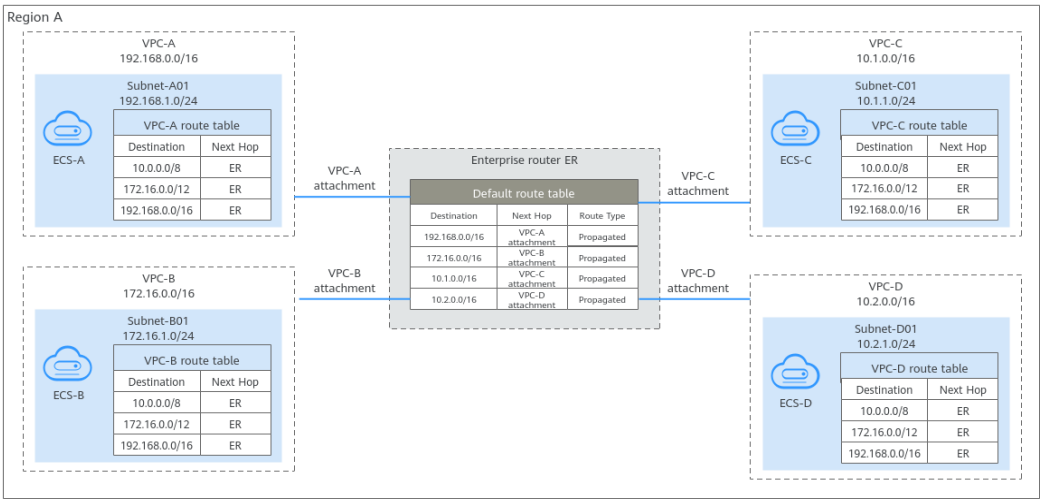
Enterprise Router

An enterprise router can connect multiple VPCs in the same account or different accounts to set up a hub-and-spoke network. Compared with VPC Peering, Enterprise Router is more suitable for complex networking where many VPCs need to be connected.

For details, see **Using an Enterprise Router to Enable Communications Between VPCs in the Same Region**.

In **Figure 2-10**, an enterprise router connects multiple VPCs in the same region and forwards traffic among them. The routes are automatically configured for the VPCs and the enterprise router.

Figure 2-10 Connecting VPCs in the same region using an enterprise router



Connecting VPCs in Different Regions

If the VPCs to be connected are located in different regions, you can use Cloud Connect, Direct Connect, or VPN.

Connecting VPCs provides details about different network services.

NOTICE

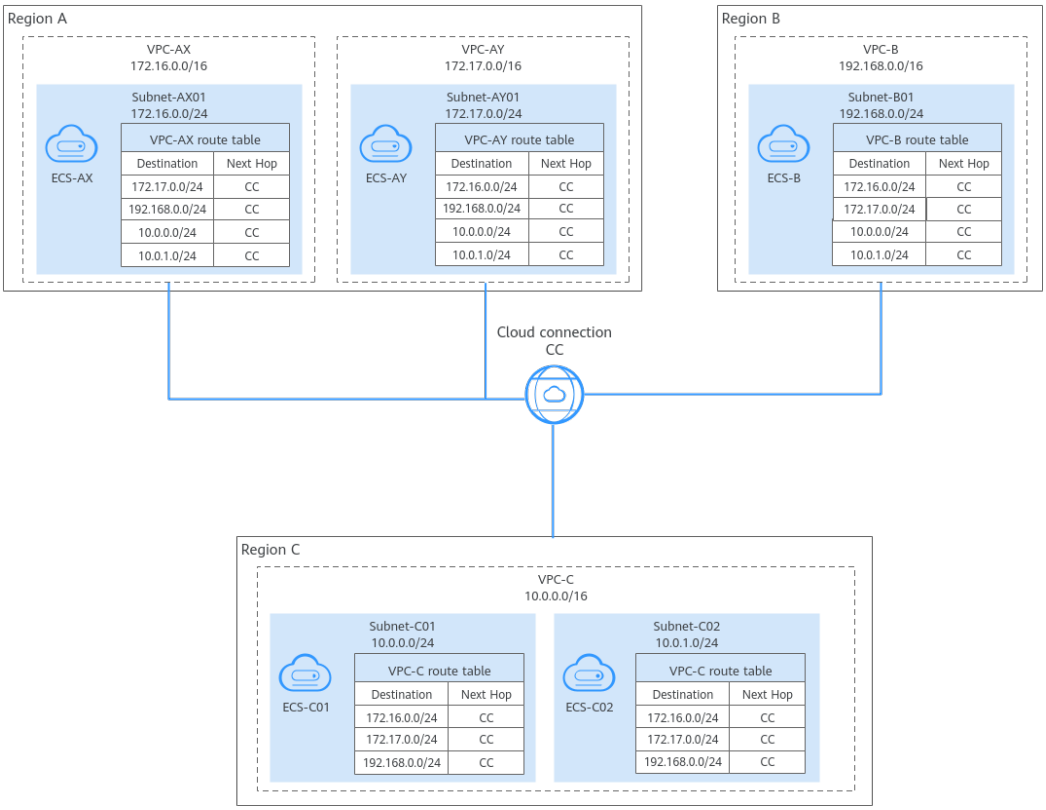
Before connecting VPCs, you need to plan their CIDR blocks in advance. Overlapping CIDR blocks may cause communication failure.

Cloud Connection

You can load VPCs in different regions to a cloud connection, regardless of whether the VPCs are in the same account or different accounts. For details, see **Connecting VPCs in Different Regions**.

In **Figure 2-11**, two VPCs (VPC-AX and VPC-AY) in region A, a VPC (VPC-B) in region B, and a VPC (VPC-C) in region C are connected over a cloud connection for private network communications.

Figure 2-11 Using a cloud connection to connect VPCs in different regions



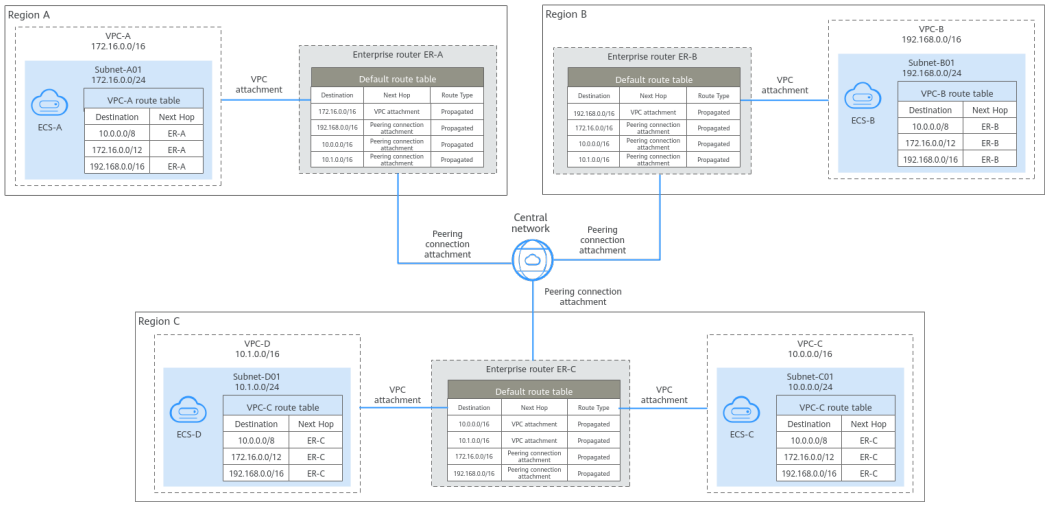
Central Network

You can attach VPCs in the same region to an enterprise router, and then add enterprise routers in different regions to a central network as attachments, so the VPCs can communicate with each other. This solution features higher scalability and is suitable for complex networking if there are multiple VPCs in different regions.

For details, see **Connecting VPCs Across Regions Using Enterprise Router and Central Network**.

In **Figure 2-12**, there are four VPCs in three regions: VPC-A in region A, VPC-B in region B, and VPC-C and VPC-D in region C. There is an enterprise router in each region: ER-A for VPC-A, ER-B for VPC-B, and ER-C for VPC-C and VPC-D. The VPCs are attached to the enterprise router in each region, and the enterprise routers in the three regions are added to a central network for cross-region network connectivity. If there will be more VPCs in the future, you only need to attach the VPCs to the enterprise router in the same region. Compared with a cloud connection, this solution features simpler network topology.

Figure 2-12 Connecting VPCs in different regions using a central network

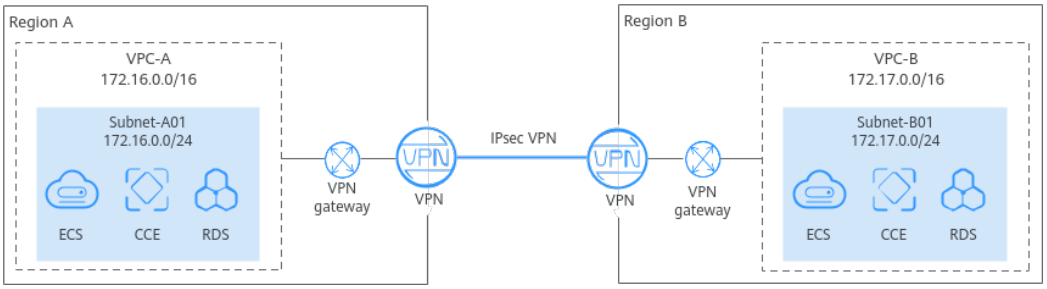


VPN

You can use **VPN** connect VPCs in different regions, so that they can communicate with each other over the Internet.

In **Figure 2-13**, there is a VPC in each region: VPC-A in region A and VPC-B in region B. Each VPC is connected to a VPN connection. The two VPCs can communicate with each other through an encrypted channel on the Internet. VPN can be enabled fast and is cost-effective.

Figure 2-13 Connecting VPCs in different regions using VPN

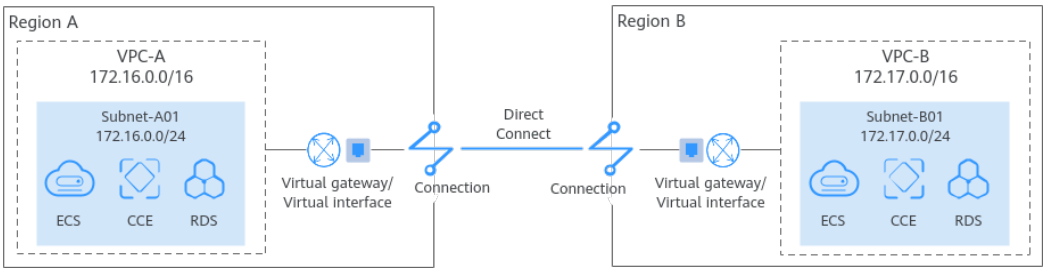


Direct Connect

You can use **Direct Connect** to connect VPCs in different regions.

In **Figure 2-14**, there is a VPC in each region: VPC-A in region A and VPC-B in region B. Each VPC is connected to a Direct Connect connection. The two VPCs can communicate with each other through a dedicated connection. Compared with VPN, Direct Connect enables faster, more stable data transmission.

Figure 2-14 Connecting VPCs in different regions using Direct Connect



2.2.3 Connecting VPCs to the Public Network

You can use EIP, NAT Gateway, or ELB to allow the resources in VPCs to access the public network.

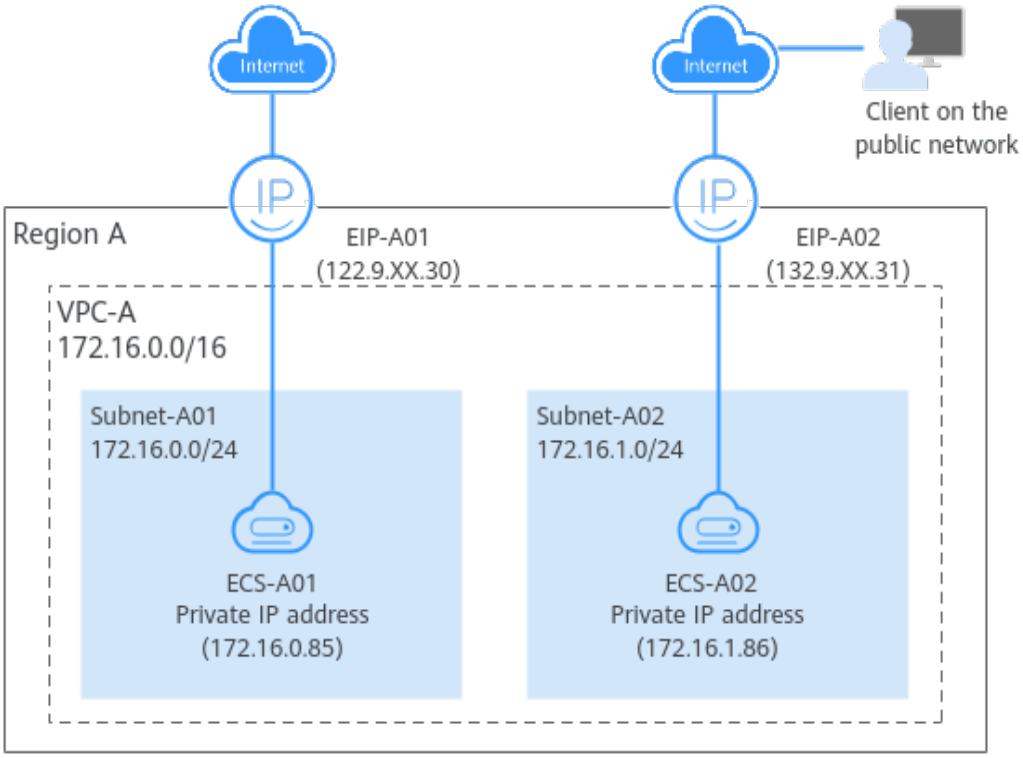
EIP

An EIP is an independent public IP address. You can bind it to an instance, such as an ECS, a NAT gateway, or a load balancer, so that the instance can access the public network or provide services accessible from the public network.

- For details about EIPs in IPv4 networks, see **Setting Up an IPv4 Network in a VPC**.
- For details about IPv4/IPv6 dual-stack networks, see **Quickly Setting Up an IPv4/IPv6 Dual-Stack Network in a VPC**.

In **Figure 2-15**, there are two subnets (Subnet-A01 and Subnet-A02) in a region (region A), and there is an ECS on each subnet. The ECS (ECS-A01) on Subnet-A01 needs to access the public network, and the ECS (ECS-A02) on Subnet-A02 needs to provide web services for the public network. Two EIPs (EIP-A01 and EIP-A02) are required, with each bound to an ECS.

Figure 2-15 Connecting a VPC to the public network using EIP



NAT Gateway (SNAT)

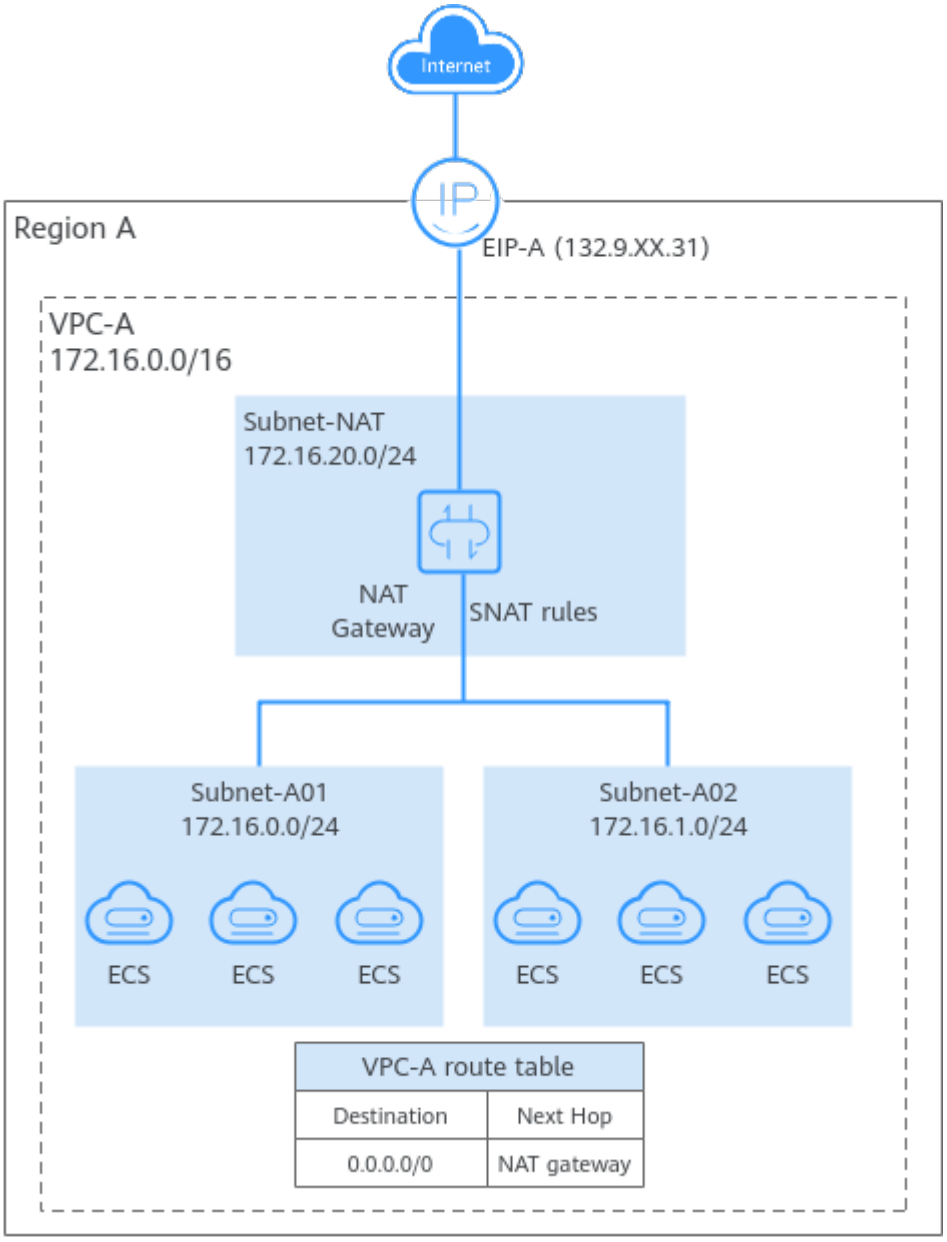
You can use a public network NAT gateway and configure SNAT rules to enable multiple ECSs in a VPC to share one or more EIPs to access the public network. If only SNAT rules are configured, the public network address of the NAT gateway cannot be directly accessed from the public network. This is more secure than using EIPs.

- If you want ECSs in a VPC to share an EIP, see **Using a Public NAT Gateway to Enable Servers to Share One or More EIPs to Access the Internet**
- If you want ECSs in different VPCs to share an EIP, see **Allowing VPCs to Share an EIP to Access the Internet Using Enterprise Router and SNAT**.

ECSs in a VPC Sharing an EIP

In **Figure 2-16**, ECSs deployed on two subnets (Subnet-A01 and Subnet-A02) in a VPC (VPC-A) need to access the public network. For this to work, you first need to create a public NAT gateway in a third subnet (Subnet-NAT), and then configure SNAT rules on the public NAT gateway for Subnet-A01 and Subnet-A02. In this way, all ECSs in Subnet-A01 and Subnet-A02 can share an EIP to access the public network.

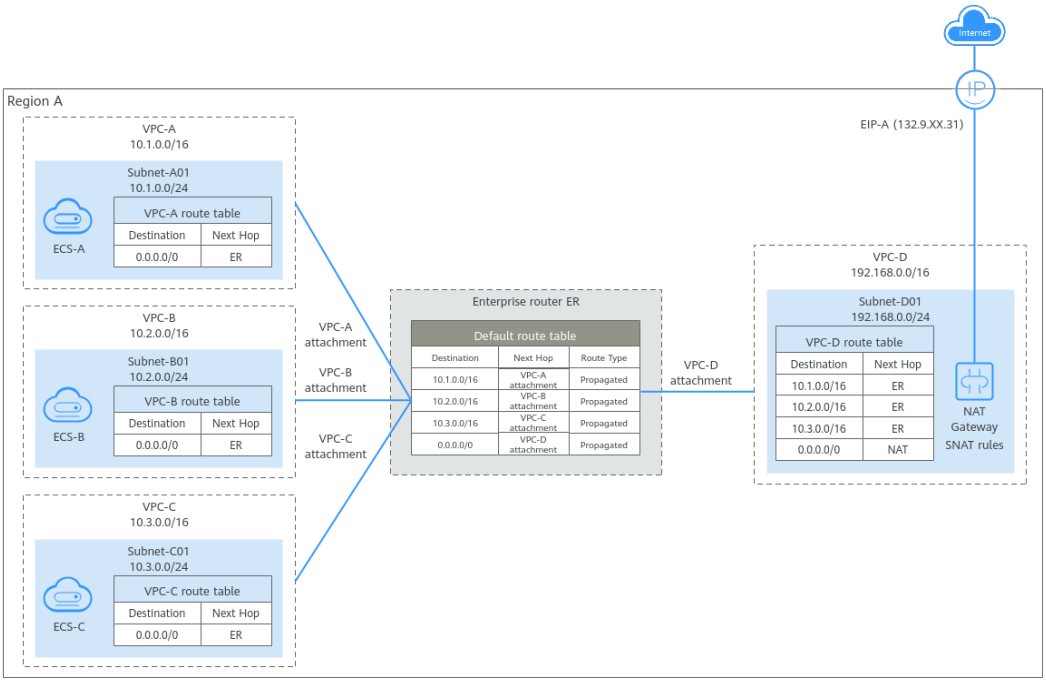
Figure 2-16 Enabling ECSs in a VPC to access the public network using a NAT gateway



ECSs in Different VPCs Sharing an EIP

In **Figure 2-17**, three VPCs (VPC-A, VPC-B, and VPC-C) in a region need to communicate with each other and can use the NAT gateway deployed in another VPC (VPC-D) to access the public network. For this to work, you first need to attach the four VPCs to an enterprise router, then configure routes in the route tables of the VPCs and of the enterprise router, and configure SNAT rules on the public NAT gateway. In this way, the VPCs can communicate with each other and share an EIP to access the public network.

Figure 2-17 Enabling ECSs in different VPCs to access the public network using a NAT gateway



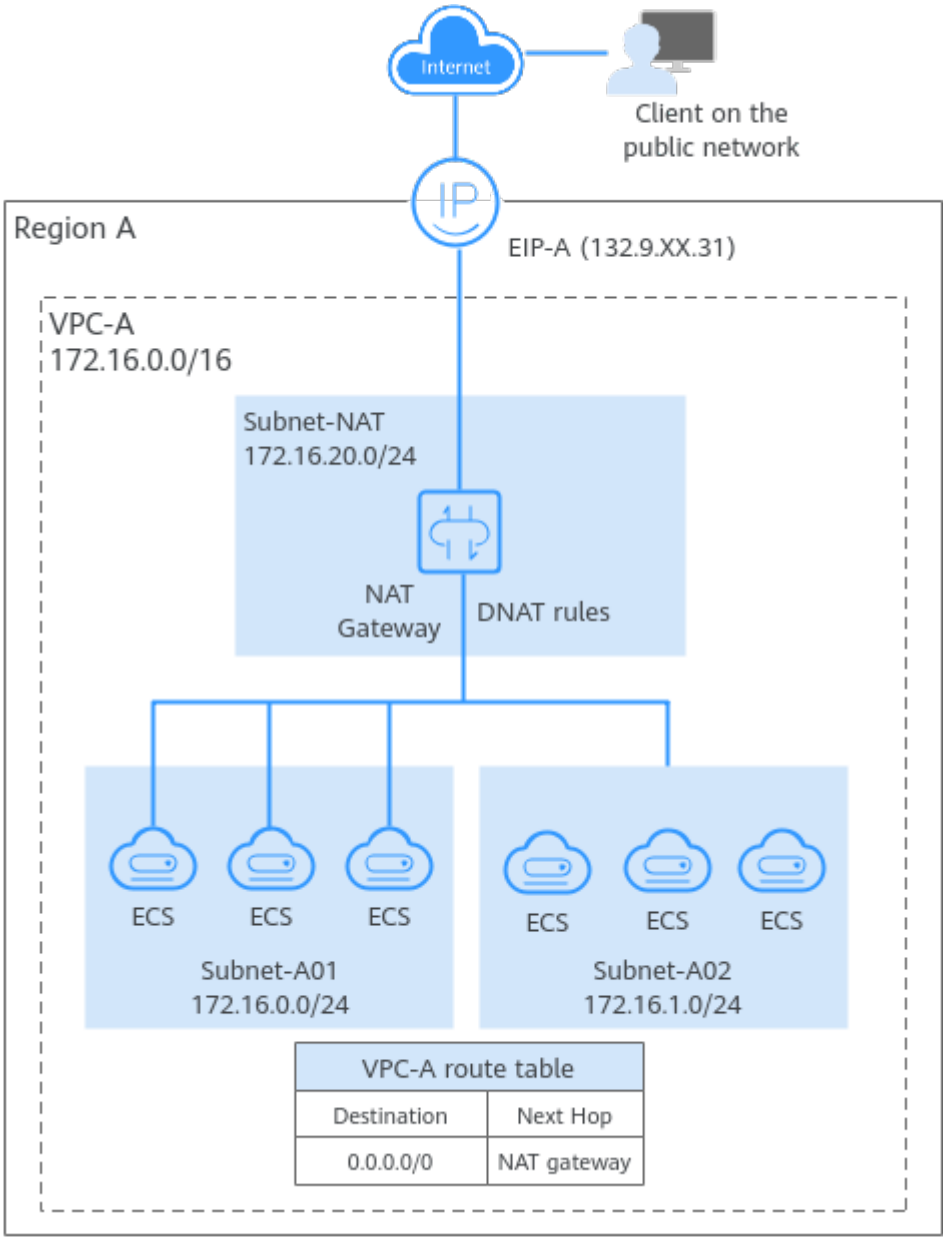
NAT Gateway (DNAT)

DNAT enables port forwarding. It maps EIP ports to ECS ports so that the ECSs in VPCs can share the same EIP and bandwidth to provide Internet-accessible services. However, DNAT does not balance traffic.

For details, see **Using a Public NAT Gateway to Enable Servers to Be Accessed by the Internet**.

In **Figure 2-18**, ECSs deployed on two subnets (Subnet-A01 and Subnet-A02) in a VPC (VPC-A) need to provide web services for the public network. For this to work, you first need to create a public NAT gateway in a third subnet (Subnet-NAT in this example), and then configure DNAT rules on the public NAT gateway for Subnet-A01 and Subnet-A02. In this way, all ECSs in Subnet-A01 and Subnet-A02 can share an EIP to provide Internet-accessible services.

Figure 2-18 Enabling ECSs in a VPC to provide services for the public network using a NAT gateway



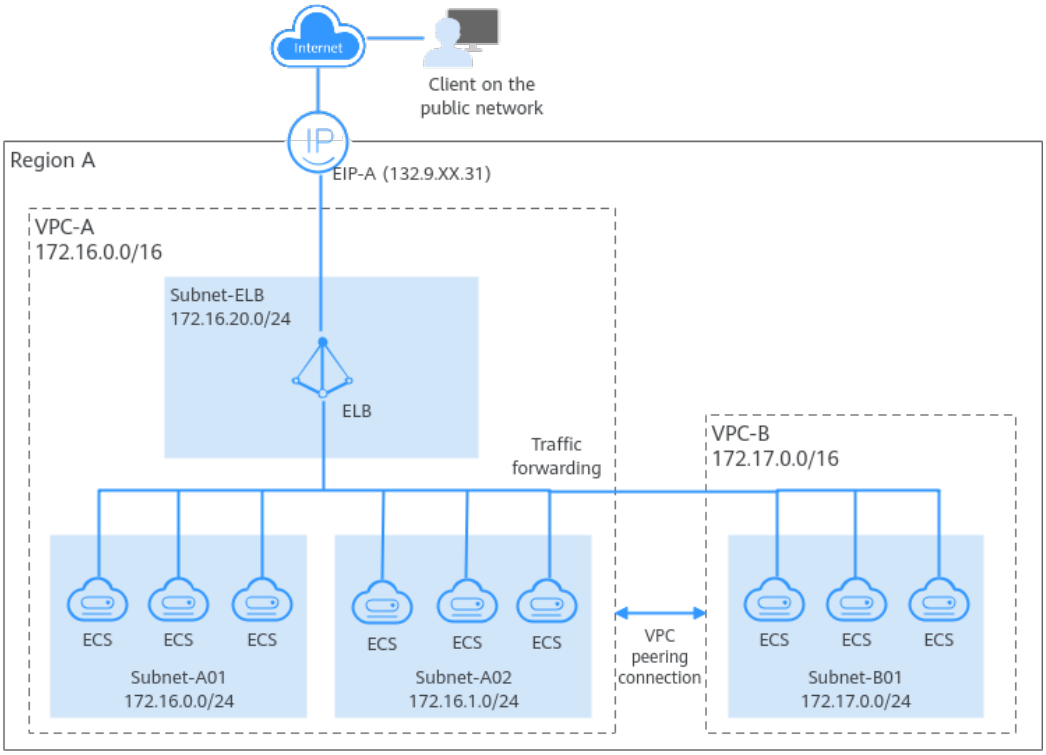
ELB

ELB evenly distributes incoming traffic to multiple backend servers. Together with EIPs, ELB allows a large number of users to access services deployed on cloud servers from the public network.

For details, see **Getting Started with ELB**.

In **Figure 2-19**, a web application is deployed on the ECSs in two VPCs (VPC-A and VPC-B) in a region. Because of the heavy incoming traffic, a load balancer is used to distribute the traffic across ECSs in different VPCs. For this to work, VPCs need to communicate with each other. In this example, a VPC peering connection is used to connect VPC-A and VPC-B.

Figure 2-19 ELB for evenly distributing incoming traffic from the public network



2.2.4 Connecting VPCs to On-Premises Data Centers

Connecting a Single VPC to an On-Premises Data Center

You can use Direct Connect or VPN to connect a VPC to an on-premises data center.

Connecting VPCs to an On-Premises Data Center provides details about different network services.

NOTICE

Before connecting a VPC to an on-premises data center, you need to plan their CIDR blocks in advance to ensure that the VPC CIDR block does not overlap with the on-premises CIDR block, or communications may fail.

VPN

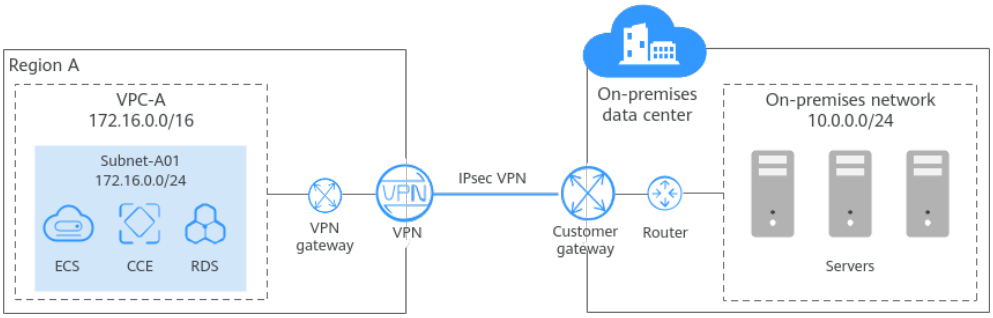
VPN provides an encrypted, Internet-based channel that connects an on-premises data center and the cloud.

For details, see **Configuring Enterprise Edition S2C VPN to Connect an On-premises Data Center to a VPC**.

In **Figure 2-20**, some workloads have been migrated to a VPC (VPC-A), and some workloads are still running on on-premises servers. With a VPN connection, on-

premises servers can quickly access the cloud resources in the VPC. Compared with Direct Connect, VPN is easier to configure and cost-effective.

Figure 2-20 Connecting a VPC to an on-premises data center using VPN



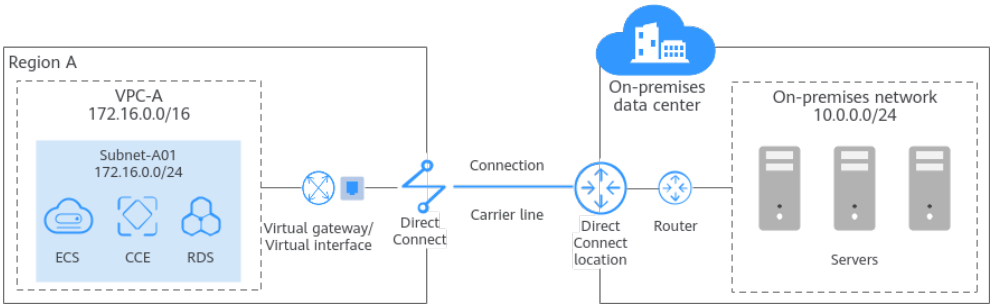
Direct Connect

Direct Connect establishes a dedicated network connection between an on-premises data center and the cloud.

For details, see **Accessing a VPC over a Direct Connect Connection and Using BGP to Route Traffic**.

In **Figure 2-21**, some workloads are running in a VPC (VPC-A) on the cloud, and some are running in the on-premises data center. A Direct Connect connection connects the on-premises data center to the cloud. Direct Connect connections are faster and more stable than VPN connections.

Figure 2-21 Connecting a VPC to an on-premises data center using Direct Connect



Connecting Multiple VPCs in the Same Region to an On-Premises Data Center

To connect multiple VPCs in a region to an on-premises data center, you can use Direct Connect or VPN to connect the data center to a VPC, and then use VPC Peering or Enterprise Router to connect all VPCs. In this way, the on-premises data center can access all the VPCs.

Compared with VPN, Direct Connect establishes a dedicated connection that enables faster, more secure data transmission. VPN is more cost-effective. To reduce network costs, you can use VPN instead of Direct Connect. **Connecting VPCs to an On-Premises Data Center** provides details about different network services.

NOTICE

To connect VPCs to an on-premises data center, you need to plan their CIDR blocks in advance. Note the following:

- Ensure that the VPC CIDR blocks do not overlap with the on-premises CIDR block, or communications may fail.
- Ensure that the VPC CIDR blocks do not overlap, or communications may fail.

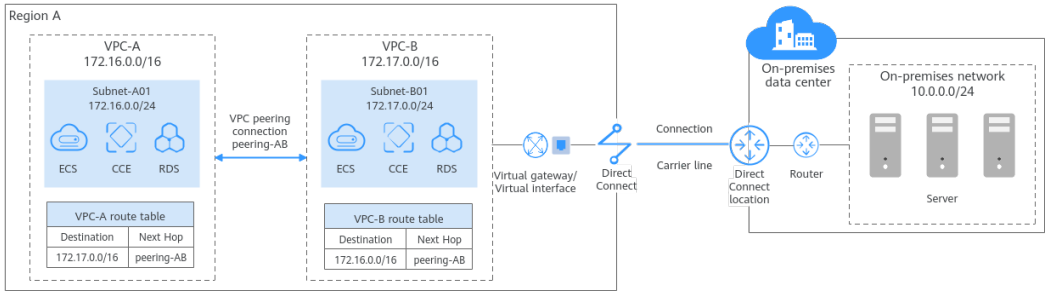
VPC Peering

With VPC Peering, you can peer two VPCs in the same region, no matter whether they are in the same account or different accounts. VPC Peering can work with Direct Connect or VPN to enable your on-premises data center to access multiple VPCs.

For details, see **Connecting an On-Premises Data Center to Multiple VPCs that Need to Communicate with Each Other**.

In **Figure 2-22**, some workloads are running in two VPCs (VPC-A and VPC-B) in a region, and some workloads are running in the on-premises data center. The on-premises data center connects to a VPC (VPC-B) over a Direct Connect connection, and VPC-A and VPC-B are connected over a VPC peering connection. In this way, the on-premises data center can access both VPC-A and VPC-B.

Figure 2-22 Connecting an on-premises data center to VPCs using Direct Connect and VPC Peering



Enterprise Router

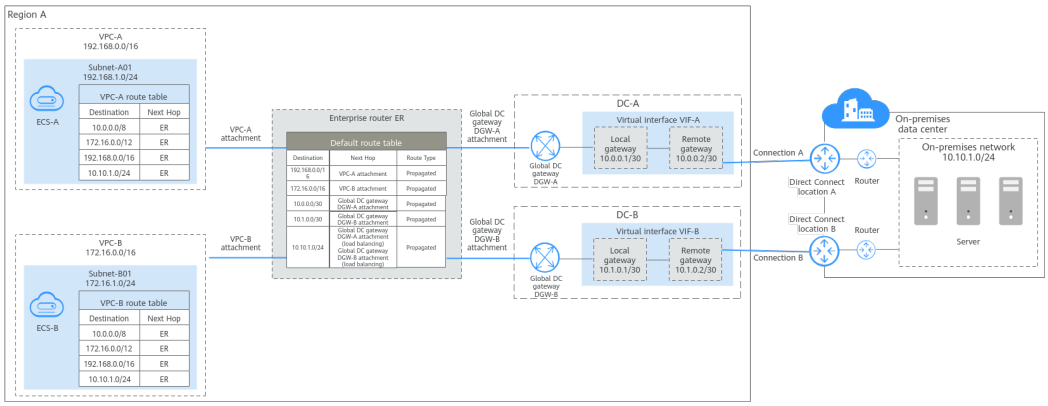
You can use VPN or Direct Connect to connect an on-premises data center to a VPC, and then use an enterprise router to connect multiple VPCs if there are in the same region.

- **Setting Up a Hybrid Cloud Network Using Enterprise Router and Direct Connect (Global DC Gateway)**
- **Setting Up a Hybrid Cloud Network Using Enterprise Router and a Pair of Direct Connect Connections (Global DC Gateway)**
- **Setting Up a Hybrid Cloud Network Using Enterprise Router and a Pair of Active/Standby Direct Connect Connections (Global DC Gateway)**
- **Setting Up a Hybrid Cloud Network Using Enterprise Router, VPN, and Direct Connect (Global DC Gateway)**

In **Figure 2-23**, some workloads are running in two VPCs (VPC-A and VPC-B) in a region, and some workloads are running in the on-premises data center. The two VPCs and global DC gateways are attached to an enterprise router in the same region, so that the two VPCs can communicate with each other and also with the on-premises data center.

In this example, two Direct Connect connections are deployed to balance loads, improving the network performance and reliability. When both connections work normally, the network transmission capability is greatly improved. If one connection becomes faulty, the other connection can take over services, and your on-premises data center can still access the VPCs.

Figure 2-23 Connecting an on-premises data center to VPCs in the same region using Direct Connect and Enterprise Router



Connecting Multiple VPCs in Different Regions to On-Premises Data Centers

To connect multiple VPCs in different regions to on-premises data centers, you can use Direct Connect or VPN to connect each on-premises data center to a VPC, and then use a cloud connection or central network to connect all VPCs.

Compared with VPN, Direct Connect establishes a dedicated connection that enables faster, more secure data transmission. VPN is more cost-effective. To reduce network costs, you can use VPN instead of Direct Connect. **Connecting VPCs to an On-Premises Data Center** provides details about different network services.

NOTICE

To connect VPCs to an on-premises data center, you need to plan their CIDR blocks in advance. Note the following:

- Ensure that the VPC CIDR blocks do not overlap with the on-premises CIDR block, or communications may fail.
- Ensure that the VPC CIDR blocks do not overlap, or communications may fail.

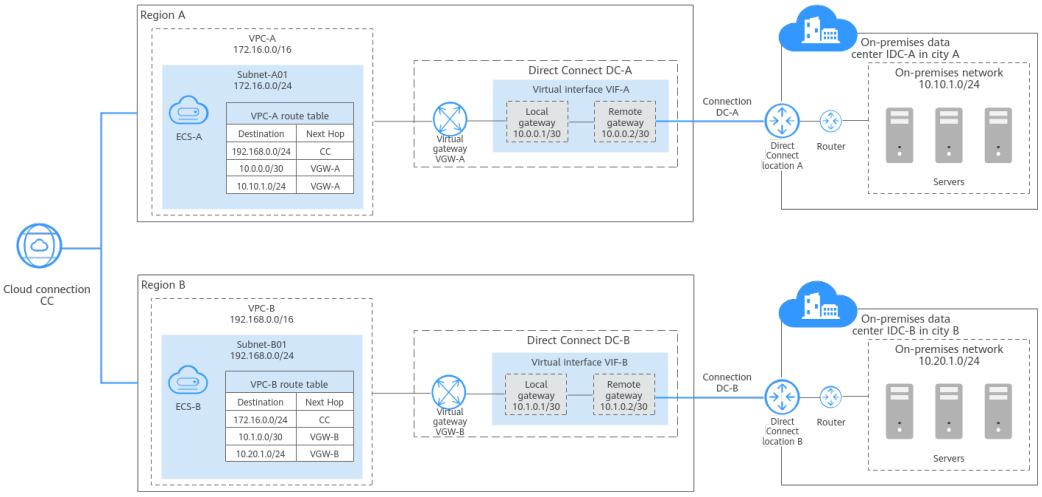
Cloud Connection

To use a cloud connection to connect VPCs in different regions, you also need to load the virtual gateway for each VPC to the cloud connection. In this way, on-premises data centers in multiple cities can access the VPCs.

For details, see **Connecting Multiple On-Premises Data Centers to Multiple VPCs in Different Regions**.

In **Figure 2-24**, there are two on-premises data centers (IDC-A and IDC-B) in different cities, with each connected to a VPC over a Direct Connect connection (DC-A and DC-B). DC-A connects IDC-A in city A to VPC-A in region A, and DC-B connects IDC-B in city B to VPC-B in region B. The two VPCs and virtual gateways are connected over a cloud connection to set up a cross-region private network. In this way, VPC-A, VPC-B, IDC-A and IDC-B can communicate with each other.

Figure 2-24 Connecting on-premises data centers to VPCs across regions using a cloud connection

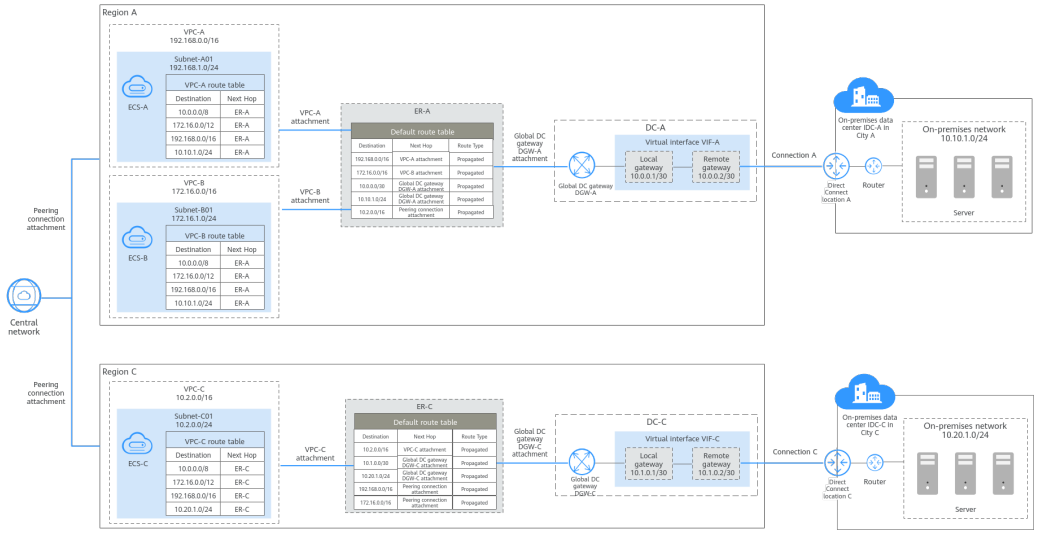


Central Network

You can attach VPCs and Direct Connect global DC gateways in the same region to an enterprise router, and then add the enterprise routers in different regions to a **central network**. In this way, VPCs in different regions can communicate with on-premises data centers in multiple cities. Compared with a cloud connection, using a central network features a simpler architecture and higher scalability.

In **Figure 2-25**, VPCs and global DC gateways in each region are attached to different enterprise routers, so the on-premises data center in each city can access the VPCs in the corresponding region. Then the two enterprise routers (ER-A and ER-C) are connected over a central network. In this way, the three VPCs (VPC-A, VPC-B, and VPC-C) and two on-premises data centers (IDC-A and IDC-C) are on the same cloud network and can communicate with each other. In this solution, only the enterprise router in each region is added to the central network, simplifying the network architecture. Also, with global DC gateways attached to enterprise routers, VPCs can share Direct Connect connections to communicate with the on-premises data centers. Route learning of enterprise routers eliminates complex configurations and simplifies maintenance.

Figure 2-25 Connecting on-premises data centers to VPCs across regions using a central network



2.3 VPC

2.3.1 Creating a VPC and Subnet

Scenarios

Virtual Private Cloud (VPC) allows you to provision logically isolated virtual private networks for cloud resources, such as cloud servers, containers, and databases.

You can create a VPC, specify a CIDR block, and create one or more subnets for the VPC. A VPC comes with a default route table that enables subnets in the VPC to communicate with each other.

Procedure

1. Go to the **Create VPC** page.
2. On the **Create VPC** page, set parameters for the VPC and subnets as prompted.

   You can click ⊕ to create more subnets. A maximum of three subnets can be created at a time.

Figure 2-26 Creating a VPC and subnet

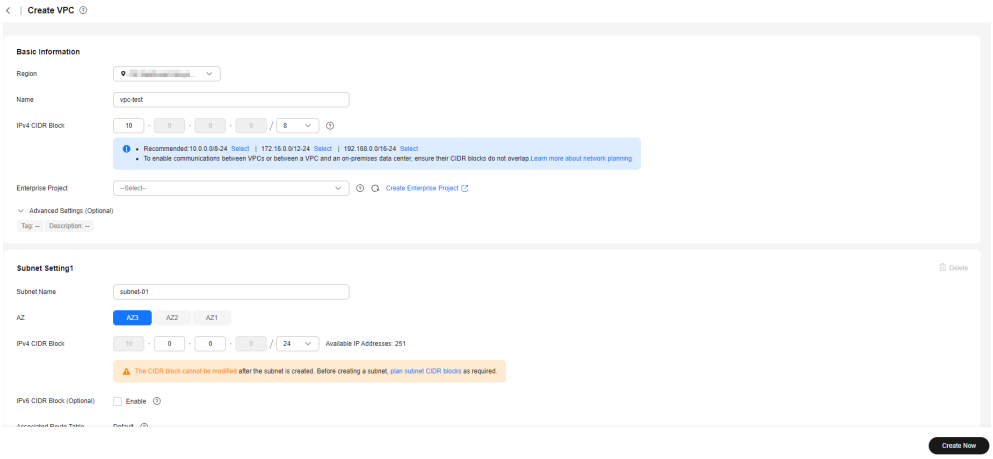


Table 2-6 VPC parameter descriptions

Parameter	Description	Example Value
Region	The region where the VPC belongs. Select the region nearest to you to ensure the lowest latency possible.	CN-Hong Kong
Name	The VPC name. The name: • Can contain 1 to 64 characters. • Can contain letters, digits, underscores (_), hyphens (-), and periods (.).	vpc-test

Parameter	Description	Example Value
IPv4 CIDR Block	The IPv4 CIDR block of the VPC. Consider the following when specifying a CIDR block: • Number of IP addresses: Reserve sufficient IP addresses for subsequent business growth. • IP address ranges: Avoid IP address conflicts if you need to connect a VPC to an on-premises data center or connect two VPCs. When you create a VPC, we recommend that you use the private IPv4 address ranges specified in **RFC 1918** as the CIDR block: • 10.0.0.0/8-24: The IP address ranges from 10.0.0.0 to 10.255.255.255, and the mask ranges from 8 to 24. • 172.16.0.0/12-24: The IP address ranges from 172.16.0.0 to 172.31.255.255, and the mask ranges from 12 to 24. • 192.168.0.0/16-24: The IP address ranges from 192.168.0.0 to 192.168.255.255, and the mask ranges from 16 to 24. In addition to the preceding addresses, you can create a VPC with a publicly routable CIDR block that falls outside of the private IPv4 address ranges specified in RFC 1918. However, the following system and public reserved addresses must be excluded: • Reserved system CIDR blocks – 100.64.0.0/10 – 214.0.0.0/7 – 198.18.0.0/15 – 169.254.0.0/16	10.0.0.0/8

Parameter	Description	Example Value
	• Reserved public CIDR blocks – 0.0.0.0/8 – 127.0.0.0/8 – 240.0.0.0/4 – 255.255.255.255/32 For details about VPC planning, see **VPC and Subnet Planning**.	
Enterprise Project	The enterprise project to which the VPC belongs. An enterprise project facilitates project-level management and grouping of cloud resources and users. The name of the default project is **default**. For details about creating and managing enterprise projects, see the **Enterprise Management User Guide**.	default
Advanced Settings (Optional) > Tag	The VPC tag. Click ⌄ to expand the configuration area and set this parameter. Add tags to help you quickly identify, classify, and search for your VPCs. For details, see **Managing VPC Tags**. **NOTE** If your organization has configured tag policies for VPCs, you need to add tags to your VPCs based on the policies. If you add a tag that does not comply with the tag policies, VPCs may fail to be created. Contact your administrator to learn more about tag policies.	• Key: vpc_key1 • Value: vpc-01

Parameter	Description	Example Value
Advanced Settings (Optional) > Description	Supplementary information about the VPC. Click to expand the configuration area and set this parameter. Enter the description about the VPC in the text box as required. The VPC description can contain a maximum of 255 characters and cannot contain angle brackets (< or >).	N/A

Table 2-7 Subnet parameter descriptions

Parameter	Description	Example Value
Subnet Name	The subnet name. The name: • Can contain 1 to 64 characters. • Can contain letters, digits, underscores (_), hyphens (-), and periods (.).	subnet-01

Parameter	Description	Example Value
AZ	An AZ is a geographic location with independent power supply and network facilities in a region. AZs are physically isolated, and AZs in the same VPC are interconnected through an internal network. Each region contains multiple AZs. If one AZ is unavailable, other AZs in the same region continue to provide services. If **Edge** is displayed, select an edge AZ based on your service requirements. If **Edge** is not displayed, you do not need to set the subnet AZ, which does not affect your service running. ● By default, all instances in different subnets of the same VPC can communicate with each other and the subnets can be in different AZs. For example, if you have a VPC with two subnets, A01 in AZ 1 and A02 in AZ 2. Subnet A01 and A02 can communicate with each other by default. ● A cloud resource and its subnet can be in different AZs. For example, a cloud server in AZ 1 can use a subnet in AZ 3. If AZ 3 becomes faulty, cloud servers in AZ 1 can still use the subnet in AZ 3, and your services are not interrupted. ● Select **Central** if you want to provision cloud resources on the cloud and run your workloads on the cloud. ● Select **Edge** if you want to provision cloud resources to an edge site and run workloads at the edge site.	AZ1

Parameter	Description	Example Value
	For details about edge sites, see **CloudPond**. For details, see **Region and AZ**. You can select an AZ for a subnet only in certain regions. See the available regions on the management console.	
CIDR Block	The CIDR block of the subnet. This parameter is displayed only in regions where IPv4/IPv6 dual stack is not supported. Set the IPv4 CIDR block of the subnet. For details, see section "IPv4 CIDR Block".	10.0.0.0/24

Parameter	Description	Example Value
IPv4 CIDR Block	The IPv4 CIDR block of the subnet. This parameter is displayed only in regions where IPv4/IPv6 dual stack is supported. A subnet is a unique CIDR block with a range of IP addresses in a VPC. Comply with the following principles when planning subnets: • Planning CIDR block size: After a subnet is created, the CIDR block cannot be changed. You need to properly plan the CIDR block in advance based on the number of IP addresses required by your service. – The subnet CIDR block size cannot be too small. Ensure that the number of available IP addresses in the subnet meets service requirements. Remember that the first and last three addresses in a subnet CIDR block are reserved for system use. For example, in subnet 10.0.0.0/24, 10.0.0.1 is the gateway address, 10.0.0.253 is the system interface address, 10.0.0.254 is used by DHCP, and 10.0.0.255 is the broadcast address. – The subnet CIDR block cannot be too large, either. If you use a CIDR block that is too large, you may not have enough CIDR blocks available for new subnets, which can be a problem when you want to scale out services. • Avoiding subnet CIDR block conflicts: Avoid CIDR block conflicts if you need to	10.0.0.0/24

Parameter	Description	Example Value
	connect two VPCs or connect a VPC to an on-premises data center. If the subnet CIDR blocks at both ends of the network conflict, **create a subnet**. A subnet mask can be between the netmask of its VPC CIDR block and /28 netmask. If a VPC CIDR block is 10.0.0.0/16, its subnet mask can between 16 to 28. For details about subnet planning, see **VPC and Subnet Planning**.	
IPv6 CIDR Block (Optional)	The IPv6 CIDR block of the subnet. This parameter is displayed only in regions where IPv4/IPv6 dual stack is supported. After the IPv6 function is enabled, the system automatically assigns an IPv6 CIDR block to the created subnet. Currently, the IPv6 CIDR block cannot be customized. IPv6 cannot be disabled after the subnet is created. For details, see **IPv4 and IPv6 Dual-Stack Network**.	-

Parameter	Description	Example Value
Associated Route Table	The default route table with which the subnet will be associated. A route table contains a set of routes that are used to control the traffic routing for your subnets in a VPC. Each VPC comes with a default route table that will be automatically associated with subnets. This allows subnets in a VPC to communicate with each other. If the default route table cannot meet your requirements, you can create a custom route table and associate subnets with it. Then, the default route table controls inbound traffic to the subnets, while the custom route table controls outbound traffic from the subnets. For details, see **Creating a Custom Route Table**.	-
Advanced Settings (Optional) > Gateway	The gateway address of the subnet. Click ⌄ to expand the configuration area and set this parameter. Retain the default value unless there are special requirements.	10.0.0.1

Parameter	Description	Example Value
Advanced Settings (Optional) > DNS Server Address	The DNS server addresses. Click ⌄ to expand the configuration area and set this parameter. Huawei Cloud private DNS server addresses are entered by default. This allows ECSs in a VPC to communicate with each other and also access other cloud services using private domain names without exposing their IP addresses to the Internet. You can change the default DNS server addresses if needed. This may interrupt your access to cloud services. You can also click **Reset** on the right to restore the DNS server addresses to the default value. A maximum of two DNS server IP addresses can be configured. Multiple IP addresses must be separated using commas (,).	100.125.x.x

Parameter	Description	Example Value
Advanced Settings (Optional) > Domain Name	The domain name. Click to expand the configuration area and set this parameter. Enter domain names (), separated with spaces. A maximum of 254 characters are allowed. A domain name can consist of multiple labels (max. 63 characters each). To access a domain name, you only need to enter the domain name prefix. ECSs in the subnet automatically match the configured domain name suffix. If the domain names are changed, ECSs newly added to this subnet will use the new domain names. If an existing ECS in this subnet needs to use the new domain names, restart the ECS or run a command to restart the DHCP Client service or network service. **NOTE** The command for updating the DHCP configuration depends on the ECS OS. The following commands are for your reference. • Restart the DHCP Client service: **service dhcpd restart** • Restart the network service: **service network restart**	test.com

Parameter	Description	Example Value
Advanced Settings (Optional) > IPv4 DHCP Lease Time	The period during which a client can use an IP address automatically assigned by the DHCP server. Click ⌄ to expand the configuration area and set this parameter. This parameter is displayed only in regions where IPv4/IPv6 dual stack is not supported. The period during which a client can use an IP address automatically assigned by the DHCP server. After the lease time expires, a new IP address will be assigned to the client. • **Limited**: Set the DHCP lease time. The unit can be day or hour. • **Unlimited**: The DHCP lease time does not expire. After you change the DHCP lease time on the console, the change is applied automatically when the DHCP lease of an instance (such as ECS) is renewed. You can wait for the system to renew the lease or manually renew the lease. Renewing lease will not change the IP address used by the instance. If you want the new lease time to take effect immediately, manually renew the lease or restart the ECS. For details, see **How Do I Make the Changed DHCP Lease Time of a Subnet Take Effect Immediately?**	-

Parameter	Description	Example Value
Advanced Settings > IPv4 DHCP Lease Time	The period during which a client can use an IPv4 address automatically assigned by the DHCP server. Click ⌄ to expand the configuration area and set this parameter. This parameter is displayed only in regions where IPv4/IPv6 dual stack is supported. You can set the DHCP lease time of an IPv4 address. The period during which a client can use an IP address automatically assigned by the DHCP server. After the lease time expires, a new IP address will be assigned to the client. • **Limited**: Set the DHCP lease time. The unit can be day or hour. • **Unlimited**: The DHCP lease time does not expire. If the time period is changed, the new lease time takes effect when the instance (such as an ECS) in the subnet is renewed next time. You can wait for the instance to be renewed automatically or manually modify the lease time. If you want the new lease time to take effect immediately, manually renew the lease or restart the ECS. For details, see **How Do I Make the Changed DHCP Lease Time of a Subnet Take Effect Immediately?**	-

Parameter	Description	Example Value
Advanced Settings > IPv6 DHCP Lease Time	The period during which a client can use an IPv6 address automatically assigned by the DHCP server. Click ⌄ to expand the configuration area and set this parameter. This parameter is displayed in the region where the IPv4/IPv6 dual stack is supported and when **IPv6** is enabled. You can set the DHCP lease time of an IPv6 address in the same way as how you do with an IPv4 address.	-
Advanced Settings (Optional) > NTP Server Address	The IP address of the NTP server. Click ⌄ to expand the configuration area and set this parameter. If you want to add NTP server addresses for a subnet, you can specify **NTP Server Address**. The IP addresses are added in addition to the default NTP server addresses. ● If you add or change the NTP server addresses of a subnet, you need to renew the DHCP lease for or restart all the ECSs in the subnet to make the change take effect immediately. ● If the NTP server addresses have been cleared out, restarting the ECSs will not help. You must renew the DHCP lease for all ECSs to make the change take effect immediately.	192.168.2.1

Parameter	Description	Example Value
Advanced Settings (Optional) > Tag	The subnet tag. Click ⌄ to expand the configuration area and set this parameter. Add tags to help you quickly identify, classify, and search for your subnets. For details, see **Managing Subnet Tags**. **NOTE** If you have configured tag policies for subnets, you need to add tags to your subnets based on the tag policies. If you add a tag that does not comply with the tag policies, subnets may fail to be created. Contact the administrator to learn more about tag policies.	• Key: subnet_key1 • Value: subnet-01
Advanced Settings (Optional) > Description	Supplementary information about the subnet. Click ⌄ to expand the configuration area and set this parameter. Enter the description about the subnet in the text box as required. The subnet description can contain a maximum of 255 characters and cannot contain angle brackets (< or >).	N/A

3. Click **Create Now**.

   Return to the VPC list and view the new VPC.

Follow-up Operations

After the VPC and subnets are created, you need to create other cloud resources in the subnets. For details, see **Setting Up an IPv4/IPv6 Dual-Stack Network In a VPC** and **Setting Up an IPv4/IPv6 Dual-Stack Network In a VPC**.

2.3.2 Adding a Secondary IPv4 CIDR Block to a VPC

Scenarios

When you create a VPC, you specify a primary IPv4 CIDR block for the VPC, which cannot be changed. To extend the IP address range of your VPC, you can add a secondary CIDR block to the VPC.

NOTE

If the **secondary IPv4 CIDR block** function is available in a region, the CIDR block of a VPC in this region cannot be modified through the console. You can call an API to modify VPC CIDR block. For details, see **Updating VPC Information**.

Notes and Constraints

- You can allocate a subnet from either a primary or a secondary CIDR block of a VPC. A subnet cannot use both the primary and the secondary CIDR blocks.

  Subnets in the same VPC can communicate with each other by default, even if some subnets are allocated from the primary CIDR block and some are from the secondary CIDR block of a VPC.

- If a subnet in a secondary CIDR block of your VPC is the same as or overlaps with the destination of an existing route in the VPC route table, the existing route does not take effect.

  If you create a subnet in a secondary CIDR block of your VPC, a route (the destination is the subnet CIDR block and the next hop is **Local**) is automatically added to your VPC route table. This route allows communications within the VPC and has a higher priority than any other routes in the VPC route table. For example, if a VPC route table has a route with the VPC peering connection as the next hop and 100.20.0.0/24 as the destination, and a route for the subnet in the secondary CIDR block has a destination of 100.20.0.0/16, 100.20.0.0/16 and 100.20.0.0/24 overlaps and traffic will be forwarded through the route of the subnet.

- The allowed secondary CIDR block size is between a /28 netmask and /3 netmask.

- **Table 2-8** lists the secondary CIDR blocks that are not supported. If 192.168.0.0/16-192.168.255.255/32 is not supported, then all CIDR blocks in this range cannot be used as secondary CIDR blocks, such as 192.168.0.0/16, 192.168.31.0/24, 192.168.100.0/24, and 192.168.255.255/32.

Table 2-8 Restricted secondary CIDR blocks

Type	CIDR Block (Not Supported)
Reserved private CIDR blocks	• 172.31.0.0/16~172.31.255.255/32 • 192.168.0.0/16~192.168.255.255/32 • In-use primary CIDR blocks
Reserved system CIDR blocks	• 100.64.0.0/10~100.127.255.255/32 • 214.0.0.0/7~215.255.255.255/32 • 198.18.0.0/15~198.19.255.255/32 • 169.254.0.0/16~169.254.255.255/32
Reserved public CIDR blocks	• 0.0.0.0/8~0.255.255.255/32 • 127.0.0.0/8~127.255.255.255/32 • 240.0.0.0/4~255.255.255.255/32

Procedure

1. Log in to the management console.
2. Click ☰ in the upper left corner and choose **Networking** > **Virtual Private Cloud**.

   The **Virtual Private Cloud** page is displayed.
3. In the VPC list, locate target VPC and click **Edit CIDR Block** in the **Operation** column.

   The **Edit CIDR Block** dialog box is displayed.
4. Click **Add Secondary IPv4 CIDR Block**.
5. Enter a secondary CIDR block and click **OK**.

2.3.3 Obtaining a VPC ID

Scenarios

This section describes how to view and obtain a VPC ID.

If you create a VPC peering connection between two VPCs in different accounts, you need to obtain the project ID of the region that the peer VPC resides. You can recommend this section to the user of the peer VPC to obtain the project ID.

Procedure

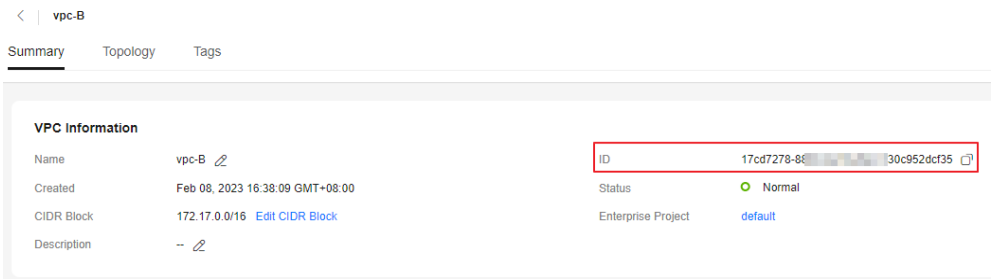
1. Log in to the management console.
2. Click 📍 in the upper left corner and select the desired region and project.
3. Click ☰ in the upper left corner and choose **Networking** > **Virtual Private Cloud**.

   The **Virtual Private Cloud** page is displayed.
4. On the **Virtual Private Cloud** page, locate the VPC and click its name.

   The VPC details page is displayed.
5. In the **VPC Information** area, view the VPC ID.

   Click ⧉ next to ID to copy the VPC ID.

Figure 2-27 VPC ID



2.3.4 Modifying a VPC

Scenarios

You can modify the following information about a VPC:

- **Modifying the Name and Description of a VPC**
- **Modifying the CIDR Block of a VPC**

> **NOTICE**
>
> If the **secondary IPv4 CIDR block** function is available in a region, the CIDR block of a VPC in this region cannot be modified through the console. You can call an API to modify VPC CIDR block. For details, see **Updating VPC Information**.

Modifying the Name and Description of a VPC

1. Log in to the management console.
2. Click in the upper left corner and select the desired region and project.
3. Click in the upper left corner and choose **Networking** > **Virtual Private Cloud**.

   The **Virtual Private Cloud** page is displayed.
4. Modify the name and description of a VPC using either of the following methods:
   - Method 1:
     i. In the VPC list, click on the right of the VPC name.
     ii. Enter a VPC name and click **OK**.
   - Method 2:
     i. In the VPC list, locate the target VPC and click its name.

     The **Summary** page is displayed.

     ii. Click on the right of the VPC name or description, enter the information, and click .

Modifying the CIDR Block of a VPC

1. Log in to the management console.
2. Click in the upper left corner and select the desired region and project.
3. Click in the upper left corner and choose **Networking** > **Virtual Private Cloud**.

   The **Virtual Private Cloud** page is displayed.
4. In the VPC list, locate the row that contains the VPC and click **Edit CIDR Block** in the **Operation** column.

The **Edit CIDR Block** dialog box is displayed.

5. Modify the VPC CIDR block as prompted.

NOTICE

A VPC CIDR block must be from 10.0.0.0/8-24, 172.16.0.0/12-24, or 192.168.0.0/16-24.

- If a VPC has no subnets, you can change both its network address and subnet mask.

  Figure 2-28 Modifying network address and subnet mask

- If a VPC has subnets, you only can change its subnet mask.

  Figure 2-29 Modifying subnet mask

6. Click **OK**.

2.3.5 Viewing a VPC Topology

Scenarios

This section describes how to view the topology of a VPC. The topology displays the subnets in a VPC and the ECSs in the subnets.

Procedure

1. Log in to the management console.
2. Click in the upper left corner and select the desired region and project.
3. Click in the upper left corner and choose **Networking** > **Virtual Private Cloud**.

The **Virtual Private Cloud** page is displayed.

4. In the VPC list, click the name of the VPC for which the topology is to be viewed.

   The VPC details page is displayed.

5. Click the **Topology** tab to view the VPC topology.

   The topology displays the subnets in the VPC and the ECSs in the subnets.

   You can also perform the following operations on subnets and ECSs in the topology:

   - Modify or delete a subnet.
   - Add an ECS to a subnet, bind an EIP to the ECS, and change the security group of the ECS.

2.3.6 Exporting VPCs

Scenarios

Information about all VPCs under your account can be exported as an Excel file to a local directory.

This file records the names, ID, status, CIDR blocks, and the number of subnets of your VPCs.

Procedure

1. Log in to the management console.
2. Click  in the upper left corner and select the desired region and project.
3. Click  in the upper left corner and choose **Networking** > **Virtual Private Cloud**.

   The **Virtual Private Cloud** page is displayed.

4. In the upper left corner of the VPC list, click **Export**.
   - **Export selected data to an XLSX file**: Select one or more VPCs and export information about the selected VPCs.
   - **Export all data to an XLSX file**: Export information about all the VPCs in the current region.

   The system will automatically export information about the VPCs as an Excel file to a local directory.

2.3.7 Managing VPC Tags

Scenarios

Tags help you identify, classify, and search for VPCs. You can perform the following operations to manage VPC tags:

- Add tags to a VPC.
- Modify a VPC tag.
- Delete a VPC tag.

If your organization has configured tag policies for VPCs, you need to add tags to your VPCs based on the policies. If you add a tag that does not comply with the tag policies, VPCs may fail to be created. Contact your administrator to learn more about tag policies.

For details about VPC tag requirements, see **Table 2-9**.

Table 2-9 VPC tag key and value requirements

Parameter	Requirements	Example Value
Key	• For each resource, each tag key must be unique, and each tag key can only have one tag value. • Cannot be left blank. • Can contain a maximum of 128 characters. • Can contain letters in any language, digits, spaces, underscores (_), periods (.), colons (:), equal signs (=), plus signs (+), minus signs (-), and at signs (@). • Cannot start with _sys_ or a space or end with a space.	vpc_key1
Value	• Can be left blank. • Can contain a maximum of 255 characters. • Can contain letters in any language, digits, spaces, underscores (_), periods (.), colons (:), slashes (/), equal signs (=), plus signs (+), minus signs (-), and at signs (@). • Cannot start or end with a space.	vpc-01

Notes and Constraints

Each cloud resource can have a maximum of 20 tags.

Procedure

1. Log in to the management console.
2. Click  in the upper left corner and select the desired region and project.
3. Click  in the upper left corner and choose **Networking** > **Virtual Private Cloud**.

   The **Virtual Private Cloud** page is displayed.
4. In the VPC list, locate the target VPC and click its name.

   The VPC details page is displayed.
5. On the **Tags** tab, click **Edit Tag** in the upper left corner above the tag list.

   The **Edit Tag** dialog box is displayed.

6. Perform the following operations on the tag as required:
   - Adding a tag: Click +, enter a tag key and value, and click **OK**.
   - Modifying a tag: Click × next to the target tag key or value, delete the original value, enter a new value, and click **OK**.
   - Deleting a tag: Click **Delete** next to the target tag and click **OK**.

2.3.8 Deleting a Secondary IPv4 CIDR Block from a VPC

Scenarios

If a secondary CIDR block of a VPC is no longer required, you can delete it.

- A secondary IPv4 CIDR block of a VPC can be deleted, but the primary CIDR block cannot be deleted.
- If you want to delete a secondary CIDR block that contains subnets, you need to delete the subnets first.

Procedure

1. Log in to the management console.
2. Click ☰ in the upper left corner and choose **Networking** > **Virtual Private Cloud**.

   The **Virtual Private Cloud** page is displayed.
3. In the VPC list, locate target VPC and click **Edit CIDR Block** in the **Operation** column.

   The **Edit CIDR Block** dialog box is displayed.
4. Locate the secondary CIDR block you want to delete and click **Delete** in the **Operation** column.
5. Click **OK**.

2.3.9 Deleting a VPC

Scenarios

If you no longer need a VPC, you can delete it.

NOTICE

The VPC service has multiple resources. Some are free, while some are not. For details about VPC resource pricing, see **Pricing Details**.

Notes and Constraints

If you want to delete a VPC that has subnets, custom routes, or other resources, you need to delete these resources as prompted on the console first and then delete the VPC.

Procedure

1. Log in to the management console.
2. Click ☰ in the upper left corner and choose **Networking** > **Virtual Private Cloud**.

   The **Virtual Private Cloud** page is displayed.
3. On the **Virtual Private Cloud** page, locate the row that contains the VPC to be deleted and click **Delete** in the **Operation** column.

   A confirmation dialog box is displayed.

   If your VPC is used by other resources, you need to delete these resources before deleting a VPC.
4. Enter **DELETE** as prompted and click **OK**.

2.4 Subnet

2.4.1 Creating a Subnet for an Existing VPC

Scenarios

A subnet is a unique CIDR block with a range of IP addresses in a VPC. All resources in a VPC must be deployed on subnets.

When creating a VPC, you need to create at least one subnet. If one subnet cannot meet your requirements, you can create more subnets for the VPC.

Notes and Constraints

After a subnet is created, some reserved IP addresses cannot be used. For example, in a subnet with CIDR block 192.168.0.0/24, the following IP addresses are reserved by default:

- 192.168.0.0: Network ID. This address is the beginning of the private IP address range and will not be assigned to any instance.
- 192.168.0.1: The gateway address of the subnet.
- 192.168.0.253: Reserved for the system interface. This IP address is used by the VPC for external communication.
- 192.168.0.254: DHCP service address.
- 192.168.0.255: Network broadcast address.

The preceding default IP addresses are only examples. The system will assign reserved IP addresses based on how you specify your subnet.

Procedure

1. Log in to the management console.
2. Click ◎ in the upper left corner and select the desired region and project.
3. Click ☰ in the upper left corner and choose **Networking** > **Virtual Private Cloud**.

The **Virtual Private Cloud** page is displayed.

4. In the navigation pane on the left, choose **Virtual Private Cloud** > **Subnets**.
5. Click **Create Subnet**.

   The **Create Subnet** page is displayed.
6. Set the subnet parameters as needed.

   You can click ⊕ to create more subnets. A maximum of three subnets can be created at a time.

Table 2-10 Subnet parameter descriptions

Parameter	Description	Example Value
Region	The region where VPC is located.	CN-Hong Kong
VPC	The VPC for which you want to create a subnet.	vpc-test
Subnet Name	The subnet name. The name: • Can contain 1 to 64 characters. • Can contain letters, digits, underscores (_), hyphens (-), and periods (.).	subnet-01

Parameter	Description	Example Value
AZ	An AZ is a geographic location with independent power supply and network facilities in a region. AZs are physically isolated, and AZs in the same VPC are interconnected through an internal network. Each region contains multiple AZs. If one AZ is unavailable, other AZs in the same region continue to provide services. If **Edge** is displayed, select an edge AZ based on your service requirements. If **Edge** is not displayed, you do not need to set the subnet AZ, which does not affect your service running. ● By default, all instances in different subnets of the same VPC can communicate with each other and the subnets can be located in different AZs. For example, if you have a VPC with two subnets, A01 in AZ 1 and A02 in AZ 2. Subnet A01 and A02 can communicate with each other by default. ● A cloud resource can be in a different AZ from its subnet. For example, a cloud server in AZ 1 can be in a subnet in AZ 3. If AZ 3 becomes faulty, cloud servers in AZ 1 can still use the subnet in AZ 3, and your services are not interrupted. ● Select **Central** if you want to provision cloud resources on the cloud and run your workloads on the cloud. ● Select **Edge** if you want to provision cloud resources to an edge site and run workloads at the edge site. For details about edge sites, see **CloudPond**. For details, see **Region and AZ**. You can select an AZ for a subnet only in certain regions. See the available regions on the management console.	AZ1

Parameter	Description	Example Value
CIDR Block	The CIDR block of the subnet. This parameter is displayed only in regions where IPv4/IPv6 dual stack is not supported. Set the IPv4 CIDR block of the subnet. For details, see section "IPv4 CIDR Block".	10.0.0.0/24

Parameter	Description	Example Value
IPv4 CIDR Block	The IPv4 CIDR block of the subnet. This parameter is displayed only in regions where IPv4/IPv6 dual stack is supported. A subnet is a unique CIDR block with a range of IP addresses in a VPC. Comply with the following principles when planning subnets: • Planning CIDR block size: After a subnet is created, the CIDR block cannot be changed. You need to properly plan the CIDR block in advance based on the number of IP addresses required by your service. – The subnet CIDR block size cannot be too small. Ensure that the number of available IP addresses in the subnet meets service requirements. Remember that the first and last three addresses in a subnet CIDR block are reserved for system use. For example, in subnet 10.0.0.0/24, 10.0.0.1 is the gateway address, 10.0.0.253 is the system interface address, 10.0.0.254 is used by DHCP, and 10.0.0.255 is the broadcast address. – The subnet CIDR block cannot be too large, either. If you use a CIDR block that is too large, you may not have enough CIDR blocks available later for new subnets, which can be a problem when you want to scale out services. • Avoid CIDR block conflicts if you need to connect two VPCs or connect a VPC to an on-premises data center. If the subnet CIDR blocks at both ends of the network conflict, **create a subnet**. A subnet mask can be between the netmask of its VPC CIDR block and /28 netmask. If a VPC CIDR block is 10.0.0.0/16, its subnet mask can between 16 to 28. If the VPC has a secondary CIDR block, you can select the primary or the secondary CIDR block that the subnet	10.0.0.0/24

Parameter	Description	Example Value
	will belong to based on service requirements.	
IPv6 CIDR Block (Optional)	The IPv6 CIDR block of the subnet. This parameter is displayed only in regions where IPv4/IPv6 dual stack is supported. If you select this option, the system automatically assigns an IPv6 CIDR block to the created subnet. Currently, the IPv6 CIDR block cannot be customized. IPv6 cannot be disabled after the subnet is created. For details, see **IPv4 and IPv6 Dual-Stack Network**.	N/A
Associated Route Table	The default route table with which the subnet will be associated. A route table contains a set of routes that are used to control the traffic routing for your subnets in a VPC. Each VPC comes with a default route table. Subnets in the VPC are then automatically associated with the default route table. The default route table ensures that subnets in a VPC can communicate with each other. If the default route table cannot meet your requirements, you can create a custom route table and associate subnets with it. Then, the default route table controls inbound traffic to the subnets, while the custom route table controls outbound traffic from the subnets. For details, see **Creating a Custom Route Table**.	N/A
Advanced Settings (Optional) > Gateway	The gateway address of the subnet. Click ⌄ to expand the configuration area and set this parameter. Retain the default value unless there are special requirements.	10.0.0.1

Parameter	Description	Example Value
Advanced Settings (Optional) > DNS Server Address	The DNS server addresses. Click ⌄ to expand the configuration area and set this parameter. Huawei Cloud private DNS server addresses are entered by default. This allows ECSs in a VPC to communicate with each other and also access other cloud services using private domain names without exposing their IP addresses to the Internet. You can change the default DNS server addresses if needed. This may interrupt your access to cloud services. You can also click **Reset** on the right to restore the DNS server addresses to the default value. A maximum of two DNS server IP addresses can be configured. Multiple IP addresses must be separated using commas (,).	100.125.x.x
Advanced Settings (Optional) > Domain Name	The domain name. Click ⌄ to expand the configuration area and set this parameter. Enter domain names (), separated with spaces. A maximum of 254 characters are allowed. A domain name can consist of multiple labels (max. 63 characters each). To access a domain name, you only need to enter the domain name prefix. ECSs in the subnet automatically match the configured domain name suffix. If the domain names are changed, ECSs newly added to this subnet will use the new domain names. If an existing ECS in this subnet needs to use the new domain names, restart the ECS or run a command to restart the DHCP Client service or network service. **NOTE** The command for updating the DHCP configuration depends on the ECS OS. The following commands are for your reference. • Restart the DHCP Client service: **service dhcpd restart** • Restart the network service: **service network restart**	test.com

Parameter	Description	Example Value
Advanced Settings (Optional) > NTP Server Address	The IP address of the NTP server. Click ⌄ to expand the configuration area and set this parameter. If you want to add NTP server addresses for a subnet, you can specify **NTP Server Address**. The IP addresses are added in addition to the default NTP server addresses. ● If you add or change the NTP server addresses of a subnet, you need to renew the DHCP lease for or restart all the ECSs in the subnet to make the change take effect immediately. ● If the NTP server addresses have been cleared out, restarting the ECSs will not help. You must renew the DHCP lease for all ECSs to make the change take effect immediately.	192.168.2.1

Parameter	Description	Example Value
Advanced Settings (Optional) > IPv4 DHCP Lease Time	The period during which a client can use an IP address automatically assigned by the DHCP server. Click ⌄ to expand the configuration area and set this parameter. This parameter is displayed only in regions where IPv4/IPv6 dual stack is not supported. The period during which a client can use an IP address automatically assigned by the DHCP server. After the lease time expires, a new IP address will be assigned to the client. • **Limited**: Set the DHCP lease time. The unit can be day or hour. • **Unlimited**: The DHCP lease time does not expire. If the time period is changed, the new lease time takes effect when the instance (such as an ECS) in the subnet is renewed next time. You can wait for the instance to be renewed automatically or manually modify the lease time. If you want the new lease time to take effect immediately, manually renew the lease or restart the ECS. For details, see **How Do I Make the Changed DHCP Lease Time of a Subnet Take Effect Immediately?**	N/A

Parameter	Description	Example Value
Advanced Settings > IPv4 DHCP Lease Time	The period during which a client can use an IPv4 address automatically assigned by the DHCP server. Click to expand the configuration area and set this parameter. This parameter is displayed only in regions where IPv4/IPv6 dual stack is supported. You can set the DHCP lease time of an IPv4 address. The period during which a client can use an IP address automatically assigned by the DHCP server. After the lease time expires, a new IP address will be assigned to the client. • **Limited**: Set the DHCP lease time. The unit can be day or hour. • **Unlimited**: The DHCP lease time does not expire. If the time period is changed, the new lease time takes effect when the instance (such as an ECS) in the subnet is renewed next time. You can wait for the instance to be renewed automatically or manually modify the lease time. If you want the new lease time to take effect immediately, manually renew the lease or restart the ECS. For details, see **How Do I Make the Changed DHCP Lease Time of a Subnet Take Effect Immediately?**	N/A
Advanced Settings > IPv6 DHCP Lease Time	The period during which a client can use an IPv6 address automatically assigned by the DHCP server. Click to expand the configuration area and set this parameter. This parameter is displayed in the region where the IPv4/IPv6 dual stack is supported and when **IPv6** is enabled. You can set the DHCP lease time of an IPv6 address. You can set the DHCP lease time of an IPv6 address in the same way as how you do with an IPv4 address.	N/A

Parameter	Description	Example Value
Advanced Settings (Optional) > Tag	The subnet tag. Click to expand the configuration area and set this parameter. Add tags to help you quickly identify, classify, and search for your subnets. For details, see **Managing Subnet Tags**. **NOTE** If you have configured tag policies for subnets, you need to add tags to your subnets based on the tag policies. If you add a tag that does not comply with the tag policies, subnets may fail to be created. Contact the administrator to learn more about tag policies.	• Key: subnet_key1 • Value: subnet-01
Advanced Settings > Description	Supplementary information about the subnet. Click to expand the configuration area and set this parameter. Enter the description about the subnet in the text box as required. The subnet description can contain a maximum of 255 characters and cannot contain angle brackets (< or >).	N/A

7. Click **Create Now**.
   Return to the subnet list and view the new subnet.

2.4.2 Modifying a Subnet

Scenarios

Modify the subnet name, NTP server address, and DNS server address.

Notes and Constraints

After a subnet is created, its AZ cannot be changed.

Procedure

1. Log in to the management console.
2. Click in the upper left corner and select the desired region and project.
3. Click in the upper left corner and choose **Networking** > **Virtual Private Cloud**.
   The **Virtual Private Cloud** page is displayed.
4. In the navigation pane on the left, choose **Virtual Private Cloud** > **Subnets**.

The **Subnets** page is displayed.

5. In the subnet list, locate the target subnet and click its name.

   The subnet details page is displayed.

6. On the **Summary** tab, click  on the right of the parameter to be modified and modify the parameter as prompted.

Table 2-11 Parameter descriptions

Parameter	Description	Example Value
Name	The subnet name. The name: • Can contain 1 to 64 characters. • Can contain letters, digits, underscores (_), hyphens (-), and periods (.).	Subnet
DNS Server Address	By default, two DNS server addresses are configured. You can change them as required. A maximum of two DNS server addresses are supported. Use commas (,) to separate every two addresses. Huawei Cloud private DNS server addresses are entered by default. This allows ECSs in a VPC to communicate with each other and also access other cloud services using private domain names without exposing their IP addresses to the Internet. You can change the default DNS server addresses if needed. This may interrupt your access to cloud services. You can also click **Reset** on the right to restore the DNS server addresses to the default value. A maximum of two DNS server IP addresses can be configured. Multiple IP addresses must be separated using commas (,).	100.125.x.x

Parameter	Description	Example Value
Domain Name	Enter domain names (), separated with spaces. A maximum of 254 characters are allowed. A domain name can consist of multiple labels (max. 63 characters each). To access a domain name, you only need to enter the domain name prefix. ECSs in the subnet automatically match the configured domain name suffix. If the domain names are changed, ECSs newly added to this subnet will use the new domain names. If an existing ECS in this subnet needs to use the new domain names, restart the ECS or run a command to restart the DHCP Client service or network service. **NOTE** The command for updating the DHCP configuration depends on the ECS OS. The following commands are for your reference. ● Restart the DHCP Client service: **service dhcpd restart** ● Restart the network service: **service network restart**	test.com

Parameter	Description	Example Value
DHCP Lease Time	This parameter is displayed only in regions where IPv4/IPv6 dual stack is not supported. The period during which a client can use an IP address automatically assigned by the DHCP server. After the lease time expires, a new IP address will be assigned to the client. • **Limited**: Set the DHCP lease time. The unit can be day or hour. • **Unlimited**: The DHCP lease time does not expire. If the time period is changed, the new lease time takes effect when the instance (such as an ECS) in the subnet is renewed next time. You can wait for the instance to be renewed automatically or manually modify the lease time. If you want the new lease time to take effect immediately, manually renew the lease or restart the ECS. For details, see **How Do I Make the Changed DHCP Lease Time of a Subnet Take Effect Immediately?**	-

Parameter	Description	Example Value
IPv4 DHCP Lease Time	This parameter is displayed only in regions where IPv4/IPv6 dual stack is supported. You can set the DHCP lease time of an IPv4 address. The period during which a client can use an IP address automatically assigned by the DHCP server. After the lease time expires, a new IP address will be assigned to the client. • **Limited**: Set the DHCP lease time. The unit can be day or hour. • **Unlimited**: The DHCP lease time does not expire. If the time period is changed, the new lease time takes effect when the instance (such as an ECS) in the subnet is renewed next time. You can wait for the instance to be renewed automatically or manually modify the lease time. If you want the new lease time to take effect immediately, manually renew the lease or restart the ECS. For details, see **How Do I Make the Changed DHCP Lease Time of a Subnet Take Effect Immediately?**	-
IPv6 DHCP Lease Time	This parameter is displayed in the region where the IPv4/IPv6 dual stack is supported and when **IPv6** is enabled. You can set the DHCP lease time of an IPv6 address. You can set the DHCP lease time of an IPv6 address in the same way as how you do with an IPv4 address.	-

Parameter	Description	Example Value
NTP Server Address	The IP address of the NTP server. This parameter is optional. You can configure the NTP server IP addresses to be added to the subnet as required. The IP addresses are added in addition to the default NTP server addresses. If you do not specify this parameter, no additional NTP server IP addresses will be added. A maximum of four unique NTP server IP addresses can be configured. Multiple IP addresses must be separated by a comma (,). If you add or change the NTP server addresses of a subnet, you need to renew the DHCP lease for or restart all the ECSs in the subnet to make the change take effect immediately. If the NTP server addresses have been cleared out, restarting the ECSs will not help. You must renew the DHCP lease for all ECSs to make the change take effect immediately.	192.168.2.1
Description	Supplementary information about the subnet. This parameter is optional. The description can contain a maximum of 255 characters and cannot contain angle brackets (< or >).	-

2.4.3 Exporting Subnets

Scenarios

Information about all subnets under your account can be exported as an Excel file to a local directory. This file records the name, ID, VPC, CIDR block, and associated route table of each subnet.

Procedure

1. Log in to the management console.
2. Click in the upper left corner and select the desired region and project.
3. Click in the upper left corner and choose **Networking** > **Virtual Private Cloud**.

   The **Virtual Private Cloud** page is displayed.
4. In the navigation pane on the left, choose **Virtual Private Cloud** > **Subnets**.

   The **Subnets** page is displayed.
5. In the upper left corner of the subnet list, click **Export**.
   - **Export selected data to an XLSX file**: Select one or more subnets and export information about the selected subnets.
   - **Export all data to an XLSX file**: Export information about all the subnets in the current region.

   The system will automatically export information about the subnets as an Excel file to a local directory.

2.4.4 Viewing and Deleting Resources in a Subnet

Scenarios

VPC subnets have private IP addresses used by cloud resources. This section describes how to view resources that are using private IP addresses of subnets. If these resources are no longer required, you can delete them.

You can view resources, including ECSs, BMSs, network interfaces, load balancers, and NAT gateways.

NOTICE

After you delete all resources in a subnet by referring to this section, the message "Delete the resource that is using the subnet and then delete the subnet." is displayed when you delete the subnet, you can refer to **Viewing IP Addresses in a Subnet**.

Procedure

1. Log in to the management console.
2. Click in the upper left corner and select the desired region and project.
3. Click in the upper left corner and choose **Networking** > **Virtual Private Cloud**.

   The **Virtual Private Cloud** page is displayed.
4. In the navigation pane on the left, choose **Virtual Private Cloud** > **Subnets**.

   The **Subnets** page is displayed.
5. Locate the target subnet and click its name.

The subnet details page is displayed.

6. On the **Summary** page, view the resources in the subnet.
   a. In the **Resources** area in the lower part of the page, view the number of resources in the subnet. Click the number to the right of each resource to view the resources in the subnet.
   b. In the **Networking Components** area on the right of the page, view the NAT gateways in the subnet.
7. Delete resources from the subnet.

Table 2-12 Viewing and deleting resources in a subnet

Resource	Reference
ECS	You cannot jump to the target ECS from the current page. To delete an ECS from the subnet, you need to go to the ECS console, search for the target ECS in the ECS list, and delete it. 1. In the ECS list, click the ECS name. The ECS details page is displayed. 2. In the **NICs** area on the **Summary** page, view the name of the subnet associated with the ECS. 3. Confirm the information and **delete the ECS**.
BMS	You cannot jump to the target BMS from the current page. To delete a BMS from the subnet, you need to go to the BMS console, search for the target BMS in the BMS list, and delete it. 1. In the BMS list, click the BMS name. The BMS details page is displayed. 2. In the **NICs** area on the **Summary** page, view the name of the subnet associated with the BMS. 3. Confirm the information and **release the BMS**.
Load balancer	You can directly jump to the target load balancer page. 1. Click the number to the right of **Load Balancers**. The load balancer list is displayed. 2. Confirm the load balancer that you want to delete and click **Delete** in the **Operation** column. For details, see **Deleting a Load Balancer**.
Network interface	You can directly jump to the target network interface page. 1. Click the number to the right of **Network Interfaces**. The **Network Interfaces** page is displayed. 2. Confirm the network interface that you want to delete and choose **More** > **Delete** in the **Operation** column. For details, see **Deleting a Network Interface**.

Resource	Reference
NAT gateway	You can directly jump to the target NAT gateway page. 1. Click the NAT gateway name in the **Networking Components** area. The NAT gateway details page is displayed. 2. Click  to return to the NAT gateway list. 3. Locate the row that contains the NAT gateway and click **Delete** in the **Operation** column. • **Deleting or Unsubscribing from a Public NAT Gateway** • **Deleting a Private NAT Gateway**

2.4.5 Viewing IP Addresses in a Subnet

Scenarios

A subnet is an IP address range in a VPC. This section describes how to view the used IP addresses in a subnet.

- Virtual IP addresses
- Private IP addresses
  - Used by the subnet itself, such as the gateway, DHCP, and system interface.
  - Used by cloud resources, such as ECSs, RDS instances, and load balancers.

Notes and Constraints

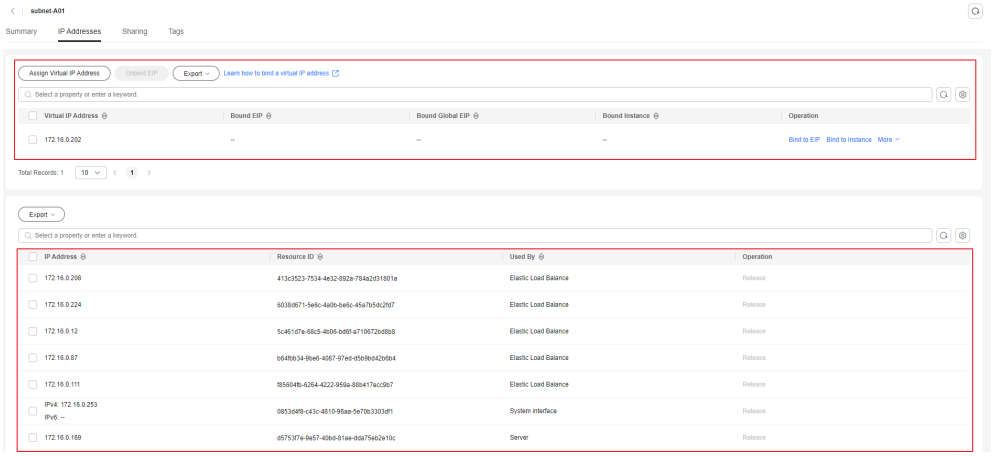
- A subnet cannot be deleted if its IP addresses are used by cloud resources.
- A subnet can be deleted if its IP addresses are used by itself.

Procedure

1. Log in to the management console.
2. Click  in the upper left corner and select the desired region and project.
3. Click  in the upper left corner and choose **Networking** > **Virtual Private Cloud**.
   The **Virtual Private Cloud** page is displayed.
4. In the navigation pane on the left, choose **Virtual Private Cloud** > **Subnets**.
   The **Subnets** page is displayed.
5. Locate the target subnet and click its name.
   The subnet details page is displayed.
6. Click the **IP Addresses** tab to view the IP addresses in the subnet.
   a. In the virtual IP address list, you can view the virtual IP addresses assigned from the subnet.

b. In the private IP address list in the lower part of the page, you can view the private IP addresses, the resources that use the IP addresses of the subnet, and the resource ID.

Figure 2-30 Viewing IP addresses in a subnet



Follow-up Operations

If you want to view and delete the resources in a subnet, refer to **Why Can't I Delete My VPCs and Subnets?**

2.4.6 Managing Subnet Tags

Scenarios

Tags help you identify, classify, and search for subnets. You can perform the following operations to manage the tags of a subnet:

- Add a tag to a subnet.
- Modify a subnet tag.
- Delete a subnet tag.

If you have configured tag policies for subnets, you need to add tags to your subnets based on the tag policies. If you add a tag that does not comply with the tag policies, subnets may fail to be created. Contact the administrator to learn more about tag policies.

For details about subnet tag requirements, see **Table 2-13**.

Table 2-13 Subnet tag key and value requirements

Parameter	Requirements	Example Value
Key	• For each resource, each tag key must be unique, and each tag key can only have one tag value. • Cannot be left blank. • Can contain a maximum of 128 characters. • Can contain letters in any language, digits, spaces, underscores (_), periods (.), colons (:), equal signs (=), plus signs (+), minus signs (-), and at signs (@). • Cannot start with _sys_ or a space or end with a space.	subnet_key1
Value	• Can be left blank. • Can contain a maximum of 255 characters. • Can contain letters in any language, digits, spaces, underscores (_), periods (.), colons (:), slashes (/), equal signs (=), plus signs (+), minus signs (-), and at signs (@). • Cannot start or end with a space.	subnet-01

Notes and Constraints

Each cloud resource can have a maximum of 20 tags.

Procedure

1. Log in to the management console.
2. Click in the upper left corner and select the desired region and project.
3. Click in the upper left corner and choose **Networking** > **Virtual Private Cloud**.

   The **Virtual Private Cloud** page is displayed.
4. In the navigation pane on the left, choose **Virtual Private Cloud** > **Subnets**.

   The **Subnets** page is displayed.
5. In the subnet list, locate the target subnet and click its name.

   The subnet details page is displayed.

6. On the **Tags** tab, click **Edit Tag** in the upper left corner above the tag list.

   The **Edit Tag** dialog box is displayed.

7. Perform the following operations on the tag as required:
   - Adding a tag: Click +, enter a tag key and value, and click **OK**.
   - Modifying a tag: Click × next to the target tag key or value, delete the original value, enter a new value, and click **OK**.
   - Deleting a tag: Click **Delete** next to the target tag and click **OK**.

2.4.7 Deleting a Subnet

Scenarios

If your subnet is no longer required, you can delete it.

NOTICE

The VPC service has multiple resources. Subnets can be used for free. For details about VPC resource pricing, see **Pricing Details**.

Notes and Constraints

If you want to delete a subnet that has custom routes, virtual IP addresses, or other resources (ECSs, load balancers, or NAT gateways), you need to delete these resources as prompted on the console first.

Procedure

1. Log in to the management console.
2. Click in the upper left corner and select the desired region and project.
3. Click in the upper left corner and choose **Networking** > **Virtual Private Cloud**.

   The **Virtual Private Cloud** page is displayed.
4. In the navigation pane on the left, choose **Virtual Private Cloud** > **Subnets**.

   The **Subnets** page is displayed.
5. In the subnet list, locate the row that contains the subnet you want to delete and click **Delete** in the **Operation** column.

   A confirmation dialog box is displayed.

   If your subnet is used by other resources, you need to delete these resources before deleting a subnet.
6. Enter **DELETE** as prompted and click **OK**.

3 Route Table and Route

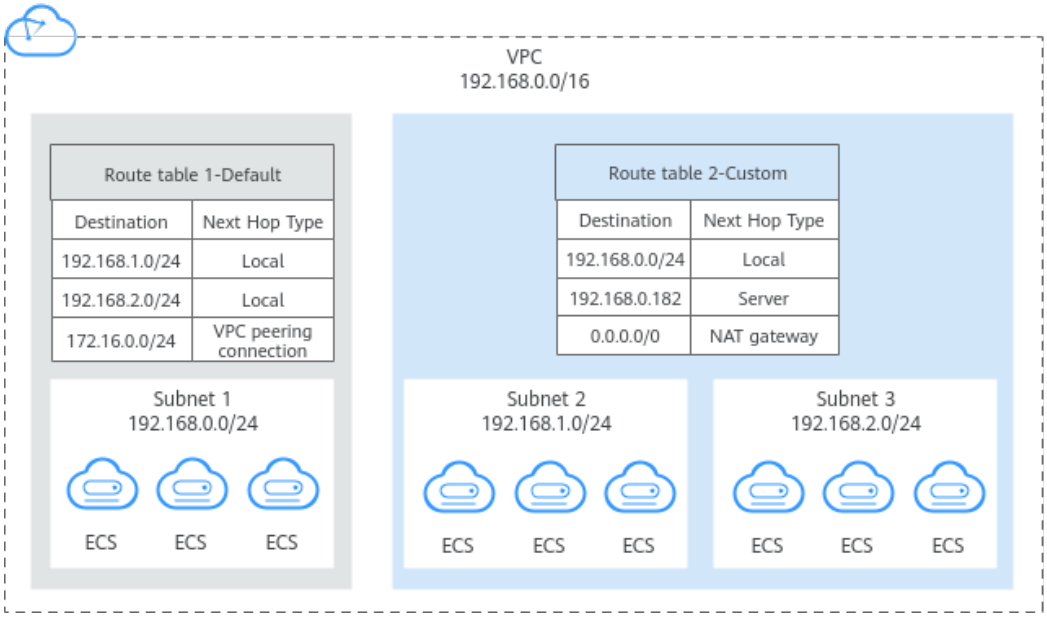
3.1 Route Table and Route Overview

What Is a Route Table?

A route table contains a set of routes that are used to control the traffic in and out of your subnets in a VPC. Each subnet must be associated with a route table. A subnet can only be associated with one route table, but a route table can be associated with multiple subnets.

Both IPv4 and IPv6 routes are supported.

Figure 3-1 Route tables



- Default route table: Each VPC comes with a default route table. If you create a subnet in a VPC, the subnet associates with the default route table. The default route table ensures that subnets in a VPC can communicate with each other.

  - You can add routes to, delete routes from, and modify routes in the default route table, but cannot delete the table.
  - When you create a VPN, Cloud Connect, or Direct Connect connection, the default route table automatically delivers a route that cannot be deleted or modified.
- Custom route table: If you do not want to use the default route table, you can create a custom route table and associate it with the subnet. Custom route tables can be deleted if they are no longer required.

  The custom route table associated with a subnet only controls the outbound traffic. The default route table of a subnet controls the inbound traffic.

NOTE

By default, the quota for custom route tables is 0. To create custom route tables, **apply for a quota increase first**.

Route

You can add routes to both default and custom route tables and configure the destination, next hop type, and next hop for the routes to determine where network traffic is directed. Routes are classified into system routes and custom routes.

- System route: A system route is automatically added by the VPC service or other services (such as VPN and Direct Connect) and cannot be deleted or modified.

  Each route table comes with routes whose next hops are Local. Generally, a route table contains the following local routes:
  - Routes whose destination is 100.64.0.0/10, which is used to deploy public services, for example, the DNS servers. The route directs instances in a subnet to access these services.
  - Routes whose destination is 198.19.128.0/20 (IP address range used by internal services, such as VPC Endpoint).
  - Routes whose destination is 127.0.0.0/8 (local loopback addresses)
  - Routes whose destination is a subnet CIDR block that enables instances in a VPC to communicate with each other.

    If you enable IPv6 when creating a subnet, the system automatically assigns an IPv6 CIDR block to the subnet. Then, you can view IPv6 routes in its route table. Example destinations of subnet CIDR blocks are as follows:

    - IPv4: 192.168.2.0/24
    - IPv6: 2407:c080:802:be7::/64
- Custom route: After a route table is created, you can add custom routes and configure information such as the destination and next hop in the route to determine where network traffic is directed. In addition to manually added custom routes, there are custom routes added by other cloud services, such as Cloud Container Engine (CCE) or NAT Gateway.

  Route tables include default route tables and custom route tables. They support the next hop types described in **Table 3-1** and **Table 3-2**. The default

route table supports fewer next hop types than a custom route table. This is because some basic services like VPN, Direct Connect, and Cloud Connect automatically add routes to the default table.

Table 3-1 Next hop types supported by the default route table

Next Hop Type	Description
Server	Traffic intended for the destination is forwarded to an ECS in the VPC.
Extension NIC	Traffic intended for the destination is forwarded to the extended network interface of an ECS in the VPC.
Supplementary network interface	Traffic intended for the destination is forwarded to the supplementary network interface of an ECS in the VPC.
NAT gateway	Traffic intended for the destination is forwarded to a NAT gateway.
VPC peering connection	Traffic intended for the destination is forwarded to a VPC peering connection.
Virtual IP address	Traffic intended for the destination is forwarded to a virtual IP address and then sent to active and standby ECSs to which the virtual IP address is bound.
VPC endpoint	Traffic intended for the destination is forwarded to a VPC endpoint.
Cloud container	Traffic intended for the destination is forwarded to a cloud container.
Enterprise router	Traffic intended for the destination is forwarded to an enterprise router.
Cloud firewall	Traffic intended for the destination is forwarded to a cloud firewall.
Global internet gateway	Traffic intended for the destination is forwarded to a global internet gateway.

Table 3-2 Next hop types supported by a custom route table

Next Hop Type	Description
Server	Traffic intended for the destination is forwarded to an ECS in the VPC.
Extension NIC	Traffic intended for the destination is forwarded to the extended network interface of an ECS in the VPC.

Next Hop Type	Description
BMS user-defined network	Traffic intended for the destination is forwarded to a BMS user-defined network.
VPN gateway	Traffic intended for the destination is forwarded to a VPN gateway.
Direct Connect gateway	Traffic intended for the destination is forwarded to a Direct Connect gateway.
Cloud connection	Traffic intended for the destination is forwarded to a cloud connection.
Supplementary network interface	Traffic intended for the destination is forwarded to the supplementary network interface of an ECS in the VPC.
NAT gateway	Traffic intended for the destination is forwarded to a NAT gateway.
VPC peering connection	Traffic intended for the destination is forwarded to a VPC peering connection.
Virtual IP address	Traffic intended for the destination is forwarded to a virtual IP address and then sent to active and standby ECSs to which the virtual IP address is bound.
VPC endpoint	Traffic intended for the destination is forwarded to a VPC endpoint.
Cloud container	Traffic intended for the destination is forwarded to a cloud container.
Enterprise router	Traffic intended for the destination is forwarded to an enterprise router.
Cloud firewall	Traffic intended for the destination is forwarded to a cloud firewall.
Global internet gateway	Traffic intended for the destination is forwarded to a global internet gateway.

NOTE

If you specify the destination when creating a resource, a system route is delivered. If you do not specify a destination when creating a resource, a custom route that can be modified or deleted is delivered.

For example, when you create a NAT gateway, the system automatically delivers a custom route without a specific destination (0.0.0.0/0 is used by default). In this case, you can change the destination. However, when you create a VPN gateway, you need to specify the remote subnet as the destination of a route. In this case, this route will be delivered as a system route. Do not modify the route destination on the **Route Tables** page. If you do, the destination will be inconsistent with the configured remote subnet. To modify the route destination, go to the specific resource page and modify the remote subnet, then the route destination will be changed accordingly.

You cannot add a route whose next hop type is **VPC endpoint** or **Cloud container** to a route table. These routes are automatically added by the VPC Endpoint or CCE service.

Notes and Constraints

When you create a VPC, the system automatically generates a default route table for the VPC. You can also create a custom route table.

- A VPC can be associated with a maximum of five route tables, including the default route table and four custom route tables.
- All route tables in a VPC can have a maximum of 1,000 routes, excluding system routes.

In each VPC route table, there are local routes and custom routes.

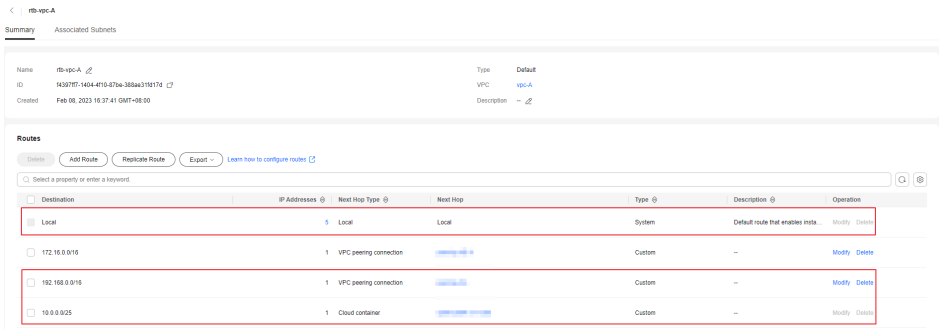
- Generally, the destination of a custom route cannot overlap with that of a local route. The destination of a local route can be a subnet CIDR block and CIDR blocks that are used for internal communications.
- You cannot add two routes with the same destination to a VPC route table even if their next hop types are different.
- When adding routes to a VPC route table, remember the route priority described in **Table 3-3**.

Table 3-3 Route priorities

Route Priority	Description
Local routes preferentially matched	A local route is the default route for communications within a VPC. They have the highest priority.

Route Priority	Description
Most accurate route (longest prefix match)	If there are multiple routes that match the request destination, the longest prefix match routing is used. This means the route that has the longest subnet mask is preferentially used to determine the next hop. Example: • A request is destined for 192.168.1.12/32. • The destination of route A is 192.168.0.0/16, with an ECS (ECS-A) as the next hop. • The destination of route B is 192.168.1.0/24, with a VPC peering connection as the next hop. According to the longest prefix match routing rule, the request preferentially matches route B and will be forwarded to the VPC peering connection.
EIP	If a custom route in the route table points to 0.0.0.0/0 and an ECS in the subnet has an EIP bound, the EIP has a higher priority. In this case, the EIP is used to access the Internet by default. Example: • The destination of route A is 0.0.0.0/0, with an NAT gateway as the next hop. • An ECS in a VPC subnet has an EIP bound. In this case, the ECS will use the EIP to access the Internet instead of the NAT gateway.

Figure 3-2 Viewing VPC route tables



Custom Route Table Configuration Process

Figure 3-3 Process for configuring a route table

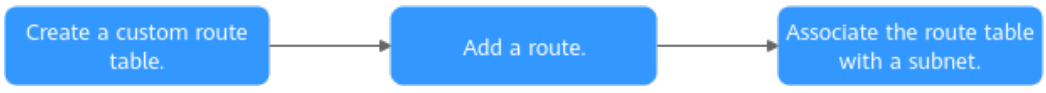


Table 3-4 Process for configuring a route table

No.	Step	Description	Reference
1	Create a custom route table.	If your default route table cannot meet your service requirements, you can create a custom route table. The custom route table associated with a subnet only controls the outbound traffic. The default route table of a subnet controls the inbound traffic.	**Creating a Custom Route Table**
2	Add a route.	You can add a custom route and configure information such as the destination and next hop in the route to determine where network traffic is directed.	**Adding Routes to a Route Table**
3	Associate the route table with a subnet.	After a route table is associated with a subnet, the routes in the route table control the routing for the subnet and apply to all cloud resources in the subnet.	**Associating a Route Table with a Subnet**

3.2 Managing Route Tables

3.2.1 Creating a Custom Route Table

Scenarios

A VPC automatically comes with a default route table. If the default route table cannot meet your service requirements, you can create a custom route table and associate subnets with it to control traffic in and out of the subnets.

Notes and Constraints

By default, the quota for custom route tables is 0. To create custom route tables, **apply for a quota increase first**.

Procedure

1. Go to the **route table list page**.
2. In the upper right corner, click **Create Route Table**. On the displayed page, configure parameters as prompted.

Table 3-5 Parameter descriptions

Parameter	Description	Example Value
Name	(Mandatory) The name of the route table. The name: • Can contain 1 to 64 characters. • Can contain letters, digits, underscores (_), hyphens (-), and periods (.).	rtb-001
VPC	(Mandatory) The VPC that the route table is used to control traffic routing. The route table can be associated with the subnets in this VPC.	vpc-001
Description	(Optional) Supplementary information about the route table. The description can contain a maximum of 255 characters and cannot contain angle brackets (< or >).	-
Route Settings	(Optional) The route information. You can add a route when creating the route table or after the route table is created. For details, see **Adding Routes to a Route Table**. You can click ⊕ to add more routes.	-

3. Click **OK**.

   A message is displayed. You can determine whether to associate the route table with subnets immediately. If you want to associate immediately, perform the following operations:

   a. Click **Associate Subnet**. The **Associated Subnets** page is displayed.
   b. Click **Associate Subnet** and select the target subnets to be associated.
   c. Click **OK**.

3.2.2 Associating a Route Table with a Subnet

Scenarios

After a subnet is created, the system associates the subnet with the default route table of its VPC. If you want to use specific routes for a subnet, you can associate the subnet with a custom route table.

The custom route table associated with a subnet affects only the outbound traffic. The default route table determines the inbound traffic.

NOTICE

After a route table is associated with a subnet, the routes in the route table control the routing for the subnet and apply to all cloud resources in the subnet.

Notes and Constraints

- A subnet must have a route table associated and can only be associated with one route table.
- A route table can be associated with multiple subnets.

Procedure

1. Log in to the management console.
2. Click  in the upper left corner and select the desired region and project.
3. Click  in the upper left corner and choose **Networking** > **Virtual Private Cloud**.

   The **Virtual Private Cloud** page is displayed.
4. In the navigation pane on the left, choose **Virtual Private Cloud** > **Route Tables**.

   The route table list is displayed.
5. In the route table list, locate the row that contains the target route table and click **Associate Subnet** in the **Operation** column.
6. Select the subnet to be associated.
7. Click **OK**.

3.2.3 Changing the Route Table Associated with a Subnet

Scenarios

You can change the route table for a subnet. If the route table is changed, routes in the new route table will apply to all cloud resources in the subnet.

Procedure

1. Log in to the management console.
2. Click  in the upper left corner and select the desired region and project.
3. Click  in the upper left corner and choose **Networking** > **Virtual Private Cloud**.

   The **Virtual Private Cloud** page is displayed.
4. In the navigation pane on the left, choose **Virtual Private Cloud** > **Route Tables**.
5. Click the name of the target route table.
6. On the **Associated Subnets** tab page, click **Change Route Table** in the **Operation** column and select a new route table as prompted.

7. Click **OK**.

   After the route table is changed, routes in the new route table will apply to all cloud resources in the subnet.

3.2.4 Viewing the Route Table Associated with a Subnet

Scenarios

You can view the route table associated with a subnet and the routes in the route table.

Procedure

1. Log in to the management console.
2. Click  in the upper left corner and select the desired region and project.
3. Click  in the upper left corner and choose **Networking** > **Virtual Private Cloud**.

   The **Virtual Private Cloud** page is displayed.
4. In the navigation pane on the left, choose **Virtual Private Cloud** > **Subnets**.

   The **Subnets** page is displayed.
5. Locate the target subnet and click its name.

   The subnet details page is displayed.
6. In the **Networking Components** area of the **Summary** page, view the route table associated with the subnet.
7. Click the name of the route table.

   The route table details page is displayed. You can further view the route information.

3.2.5 Viewing Route Table Information

Scenarios

You can view the following information about a route table:

- Basic information, such as name, type (default or custom), and ID of the route table
- Routes, such as destination, next hop, and route type (system or custom)
- Associated subnets

Procedure

1. Log in to the management console.
2. Click  in the upper left corner and select the desired region and project.
3. Click  in the upper left corner and choose **Networking** > **Virtual Private Cloud**.

   The **Virtual Private Cloud** page is displayed.

4. In the navigation pane on the left, choose **Virtual Private Cloud** > **Route Tables**.

   The route table list is displayed.

5. Click the name of the target route table.

   The route table details page is displayed.

   a. On the **Summary** tab page, view the basic information and routes of the route table.

   b. On the **Associated Subnets** tab page, view the subnets associated with the route table.

3.2.6 Deleting a Route Table

Scenarios

If you no longer need a custom route table, you can delete it.

Notes and Constraints

- The default route table cannot be deleted.

  However, deleting a VPC will also delete its default route table. Both default and custom route tables are free of charge.

- A custom route table with a subnet associated cannot be deleted directly.

  If you want to delete such a route table, you can associate the subnet with another route table first by referring to **Changing the Route Table Associated with a Subnet**.

Procedure

1. Log in to the management console.
2. Click  in the upper left corner and select the desired region and project.
3. Click  in the upper left corner and choose **Networking** > **Virtual Private Cloud**. The **Virtual Private Cloud** page is displayed.
4. In the navigation pane on the left, choose **Virtual Private Cloud** > **Route Tables**.

   The route table list is displayed.

5. Locate the row that contains the route table you want to delete and click **Delete** in the **Operation** column.

   A confirmation dialog box is displayed.

6. Confirm the information and click **OK**.

3.3 Managing Routes

3.3.1 Adding Routes to a Route Table

Scenarios

Each route table comes with a default route, which is used to allow instances in a subnet to access public services on the cloud or different subnets in a VPC to communicate with each other. You can also add custom routes as required to control traffic routing.

If a route table is associated with a subnet, adding rules to the route table may affect how and where traffic is directed. Be careful with this operation as it may interrupt services.

Notes and Constraints

- Generally, the destination of a custom route cannot overlap with that of a local route. The destination of a local route can be a subnet CIDR block and CIDR blocks that are used for internal communications.
- You cannot add two routes with the same destination to a VPC route table even if their next hop types are different.

Procedure

1. Log in to the management console.
2. Click  in the upper left corner and select the desired region and project.
3. Click  in the upper left corner and choose **Networking** > **Virtual Private Cloud**.

   The **Virtual Private Cloud** page is displayed.
4. In the navigation pane on the left, choose **Virtual Private Cloud** > **Route Tables**.

   The route table list is displayed.
5. Locate the target route table and click its name.

   The route table details page is displayed.
6. Click **Add Route** and set parameters as prompted.

   You can click  to add more routes.

Table 3-6 Parameter descriptions

Parameter	Description	Example Value
Destination Type	Mandatory The destination type can only be **IP address**. You can set an IP address or CIDR block.	IP address

Parameter	Description	Example Value
Destination	Mandatory Enter the destination of the route. You can enter a single IP address or an IP address range in CIDR notation. **NOTICE** If an IP address group contains an IP address range in the format of *Start IP address-End IP address*, the IP address group is not supported. For example, an IP address group cannot contain 192.168.0.1-192.168.0.62. You need to change 192.168.0.1-192.168.0.62 to 192.168.0.0/26.	IPv4: 192.168.0.0/16
Next Hop Type	Mandatory Set the type of the next hop. **NOTE** When you add or modify a custom route in a default route table, the next hop type of the route cannot be set to **VPN gateway**, **Direct Connect gateway**, or **Cloud connection**.	VPC peering connection
Next Hop	Mandatory Set the next hop. The resources in the drop-down list box are displayed based on the selected next hop type.	peer-AB
Description	Optional Enter the description of the route in the text box as required.	-

7. Click **OK**.
   You can view the new routes in the route list.

3.3.2 Modifying a Route

Scenarios

You can modify an existing route in a route table.

If a route table is associated with a subnet, modifying rules in the route table may affect how and where traffic is directed. Be careful with this operation as it may interrupt services.

Notes and Constraints

- System routes cannot be modified.
- When you create a VPN, Cloud Connect, or Direct Connect connection, the default route table automatically delivers a route that cannot be deleted or modified.
- Routes with the next hop type of cloud container cannot be modified or deleted.

- Routes with the next hop type of VPC endpoint cannot be modified or deleted.

Procedure

1. Log in to the management console.
2. Click  in the upper left corner and select the desired region and project.
3. Click  in the upper left corner and choose **Networking** > **Virtual Private Cloud**.

   The **Virtual Private Cloud** page is displayed.
4. In the navigation pane on the left, choose **Virtual Private Cloud** > **Route Tables**.

   The route table list is displayed.
5. Locate the target route table and click its name.

   The route table details page is displayed.
6. Locate the target route and click **Modify** in the **Operation** column.
7. Modify the route information in the displayed dialog box.

Table 3-7 Parameter descriptions

Parameter	Description	Example Value
Destination Type	Mandatory The destination type can only be **IP address**. You can set an IP address or CIDR block.	IP address
Destination	Mandatory Enter the destination of the route. You can enter a single IP address or an IP address range in CIDR notation. **NOTICE** If an IP address group contains an IP address range in the format of *Start IP address-End IP address*, the IP address group is not supported. For example, an IP address group cannot contain 192.168.0.1-192.168.0.62. You need to change 192.168.0.1-192.168.0.62 to 192.168.0.0/26.	IPv4: 192.168.0.0/16
Next Hop Type	Mandatory Set the type of the next hop. **NOTE** When you add or modify a custom route in a default route table, the next hop type of the route cannot be set to **VPN gateway**, **Direct Connect gateway**, or **Cloud connection**.	VPC peering connection

Parameter	Description	Example Value
Next Hop	Mandatory Set the next hop. The resources in the drop-down list box are displayed based on the selected next hop type.	peer-AB
Description	Optional Enter the description of the route in the text box as required.	-

8. Click **OK**.

3.3.3 Replicating a Route

Scenarios

You can replicate a route from a custom route table to one another within a VPC. You can also replicate a route from the default route table to a custom route table, or the other way around.

Notes and Constraints

Table 3-8 shows whether routes of different types can be replicated to default or custom route tables.

If the next hop type of a route is a server, this route can be replicated to both default and custom route tables.

If the next hop type of a route is a Direct Connect gateway, the route cannot be replicated to the default route table, but can be replicated to a custom route table.

Table 3-8 Route replication

Next Hop Type	Can Be Replicated to the Default Route Table	Can Be Replicated to a Custom Route Table
Local	No	No
Server	Yes	Yes
Extension NIC	Yes	Yes
BMS user-defined network	No	Yes
VPN gateway	No	Yes
Direct Connect gateway	No	Yes
Cloud connection	No	Yes

Next Hop Type	Can Be Replicated to the Default Route Table	Can Be Replicated to a Custom Route Table
Supplementary network interface	Yes	Yes
NAT gateway	Yes	Yes
VPC peering connection	Yes	Yes
Virtual IP address	Yes	Yes
VPC endpoint	No	No
Cloud container	No	No
Enterprise router	Yes	Yes
Cloud firewall	Yes	Yes

NOTE

- If the Direct Connect service is enabled by call or email, the routes delivered to the default route table cannot be replicated to a custom route table.

Procedure

1. Log in to the management console.
2. Click in the upper left corner and select the desired region and project.
3. Click in the upper left corner and choose **Networking** > **Virtual Private Cloud**.

   The **Virtual Private Cloud** page is displayed.
4. In the navigation pane on the left, choose **Virtual Private Cloud** > **Route Tables**.

   The route table list is displayed.
5. Locate the target route table and click its name.

   The route table details page is displayed.
6. Click **Replicate Route** above the route list and select the target route table and route.
7. Click **OK**.

3.3.4 Deleting a Route

Scenarios

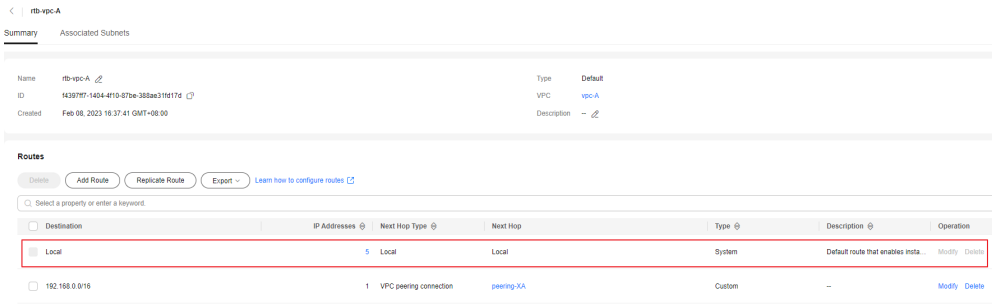
You can delete a custom route from a route table.

If a route table is associated with a subnet, deleting rules from the route table may affect how and where traffic is directed. Be careful with this operation as it may interrupt services.

Notes and Constraints

- System routes cannot be deleted.

  Figure 3-4 System routes

- The routes automatically delivered by VPN, Cloud Connect, or Direct Connect to the default route table cannot be deleted. The next hop types of such routes are:
  - VPN gateway
  - Direct Connect gateway
  - Cloud connection

  To delete these routes, you need to delete the associated network instances first.
- Routes with the next hop type of cloud container cannot be modified or deleted.
- Routes with the next hop type of VPC endpoint cannot be modified or deleted.

Procedure

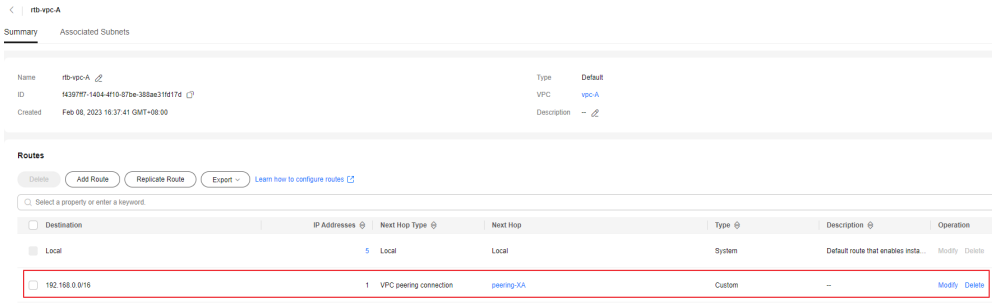
1. Log in to the management console.
2. Click  in the upper left corner and select the desired region and project.
3. Click  in the upper left corner and choose **Networking** > **Virtual Private Cloud**.

   The **Virtual Private Cloud** page is displayed.
4. In the navigation pane on the left, choose **Virtual Private Cloud** > **Route Tables**.

   The route table list is displayed.
5. Locate the target route table and click its name.

   The route table details page is displayed.

Figure 3-5 Viewing a custom route



6. In the route list, locate the row that contains the route to be deleted and click **Delete** in the **Operation** column.

   A confirmation dialog box is displayed.

7. Confirm the information and click **OK**.

3.4 Route Configuration Examples

3.4.1 Configuring an SNAT Server to Enable ECSs to Share an EIP to Access the Internet

Scenarios

Together with VPC route tables, you can configure SNAT on an ECS to enable other ECSs that have no EIPs bound in the same VPC to access the Internet through this ECS.

The configured SNAT takes effect for all subnets in a VPC.

Prerequisites

- You have an ECS where SNAT is to be configured.
- The ECS where SNAT is to be configured runs Linux.
- The ECS where SNAT is to be configured has only one network interface.

Differences Between SNAT ECSs and NAT Gateways

NAT Gateway provides network address translation (NAT) for servers, such as ECSs and BMSs, in a VPC or servers that connect to a VPC through Direct Connect or VPN in local data centers, allowing these servers to access the Internet using EIPs or to provide services for the Internet.

NAT Gateway is easier to configure and use than SNAT. This service can be flexibly deployed across subnets and AZs and has different NAT gateway specifications. You can click **NAT Gateway** under **Networking** on the management console to try this service.

For details, see the ***NAT Gateway User Guide***.

Procedure

1. Log in to the management console.
2. Click in the upper left corner and select the desired region and project.
3. In the upper left corner of the page, click . In the service list, choose **Compute** > **Elastic Cloud Server**.
4. On the displayed page, locate the target ECS in the ECS list and click its name to go to the page showing ECS details.
5. On the displayed page, click the **Network Interfaces** tab.
6. Click the network interface IP address to view details and disable **Source/Destination Check**.

   By default, the source/destination check option is enabled to check whether source IP addresses contained in the packets sent by ECSs are correct. If the IP addresses are incorrect, the system does not allow the ECSs to send the packets. This mechanism prevents packet spoofing, thereby improving system security. If the SNAT function is used, the SNAT server needs to forward packets. This mechanism prevents the packet sender from receiving returned packets. Therefore, you need to disable the source/destination check for SNAT servers.
7. Bind an EIP.
   - Bind an EIP to the private IP address of the ECS. For details, see **Binding an EIP to an Instance**.
   - Bind an EIP to the virtual IP address of the ECS. For details, see **Binding a Virtual IP Address to an Instance or EIP**.
8. On the ECS console, remotely log in to the ECS where you plan to configure SNAT.
9. Run the following command and enter the password of user **root** to switch to user **root**:

   su - root
10. Run the following command to check whether the ECS can successfully connect to the Internet:

    NOTE

    Before running the command, you must disable the response iptables rule on the ECS where SNAT is configured and configure security group rules.

    ping support.huawei.com

    The ECS can access the Internet if the following information is displayed:

```
[root@localhost ~]# ping support.huawei.com
PING support.huawei.com (xxx.xxx.xxx.xxx) 56(84) bytes of data.
64 bytes from xxx.xxx.xxx.xxx: icmp_seq=1 ttl=51 time=9.34 ms
64 bytes from xxx.xxx.xxx.xxx: icmp_seq=2 ttl=51 time=9.11 ms
64 bytes from xxx.xxx.xxx.xxx: icmp_seq=3 ttl=51 time=8.99 ms
```

11. Run the following command to check whether IP forwarding of the Linux OS is enabled:

    cat /proc/sys/net/ipv4/ip_forward

    In the command output, **1** indicates that IP forwarding is enabled, and **0** indicates that IP forwarding is disabled. The default value is **0**.

- If IP forwarding in Linux is enabled, go to step 14.
- If IP forwarding in Linux is disabled, go to 12 to enable IP forwarding in Linux.

Many OSs support packet routing. Before forwarding packets, OSs change source IP addresses in the packets to OS IP addresses. Therefore, the forwarded packets contain the IP address of the public sender so that the response packets can be sent back along the same path to the initial packet sender. This method is called SNAT. The OSs need to keep track of the packets where IP addresses have been changed to ensure that the destination IP addresses in the packets can be rewritten and that packets can be forwarded to the initial packet sender. To achieve these purposes, you need to enable the IP forwarding function and configure SNAT rules.

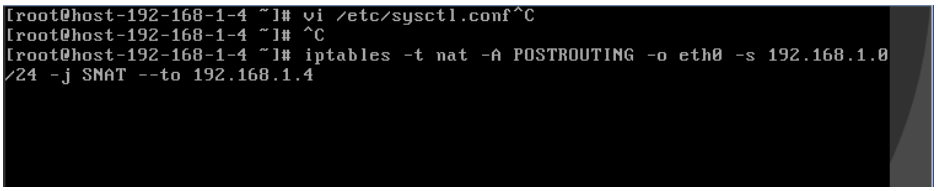
12. Use the vi editor to open the **/etc/sysctl.conf** file, change the value of **net.ipv4.ip_forward** to **1**, and enter **:wq** to save the change and exit.
13. Run the following command to make the change take effect:

    sysctl -p /etc/sysctl.conf
14. Configure the SNAT function.

    Run the following command to allow all ECSs in the subnet (for example, 192.168.1.0/24) to access the Internet: Example command:

    iptables -t nat -A POSTROUTING -o eth0 -s subnet -j SNAT --to nat-instance-ip

    Figure 3-6 Configuring SNAT

    NOTE

    To ensure that the rule will not be lost after the restart, write the rule into the **/etc/rc.local** file.

    1. Switch to the **/etc/sysctl.conf** file:

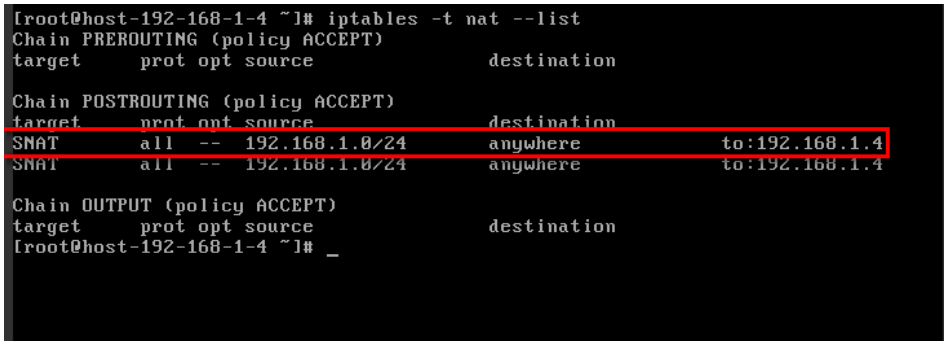
       vi /etc/rc.local
    2. Perform 14 to configure SNAT.
    3. Save the configuration and exit:

       :wq
    4. Add the execution permissions for the **rc.local** file:

       # chmod +x /etc/rc.local
15. Check whether the configuration is successful. If information similar to **Figure 3-7** (for example, 192.168.1.0/24) is displayed, the configuration was successful.

    iptables -t nat --list

Figure 3-7 Verifying configuration



16. Add a route. For details, see section **Adding Routes to a Route Table**.

    Set the destination to **0.0.0.0/0**, and the next hop to the private or virtual IP address of the ECS where SNAT is deployed. For example, the next hop is **192.168.1.4**.

After these operations are complete, if the network communication still fails, check your security group and network ACL configuration to see whether required traffic is allowed.

4 Virtual IP Address

4.1 Virtual IP Address Overview

What Is a Virtual IP Address?

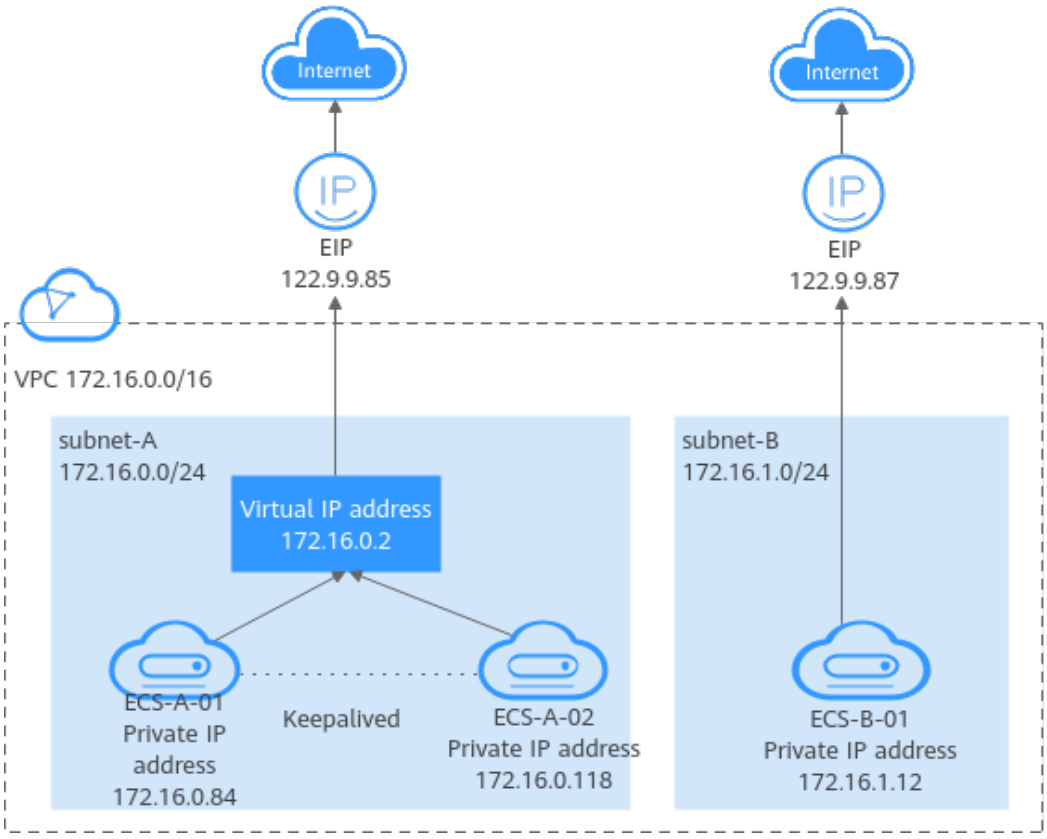
A virtual IP address is a private IP address that can be independently assigned from and released to a VPC subnet. You can:

- Bind one or more virtual IP addresses to a cloud server so that you can use either the virtual IP address or private IP address to access the server. If you have multiple services running on a cloud server, you can use different virtual IP addresses to access them.
- Bind a virtual IP address to multiple cloud servers. You can use a virtual IP address and an HA software (such as Keepalived) to set up a high-availability active/standby cluster. If you want to improve service availability and eliminate single points of failure, you can deploy cloud servers in the active/standby pair or deploy one cloud server and multiple standby cloud servers. In this case, the cloud servers can use the same virtual IP address. If the active cloud server goes down, the standby cloud server becomes the active server and continues to provide services.

Generally, cloud servers use private IP addresses for internal network communication. A virtual IP address has the same network access capabilities as a private IP address. You can use either of them to enable layer 2 and layer 3 communications in a VPC, access a different VPC using a peering connection, enable Internet access through EIPs, and connect the cloud and the on-premises servers using VPN connections and Direct Connect connections. **Figure 4-1** describes how private IP addresses, the virtual IP address, and EIPs work together.

- Private IP addresses are used for internal network communication.
- The virtual IP address works with Keepalived to build an HA cluster. ECSs in this cluster can be accessed through one virtual IP address.
- EIPs are used for Internet communication.

Figure 4-1 Different types of IP addresses used by ECSs



Application Scenarios

You can use a virtual IP address and Keepalived to set up a high-availability active/standby cluster. If the active cloud server goes down, the standby server becomes the active server and continues to provide services. The following describes the typical application scenarios of virtual IP addresses.

Using a Virtual IP Address and Keepalived to Set Up a High-Availability Cluster

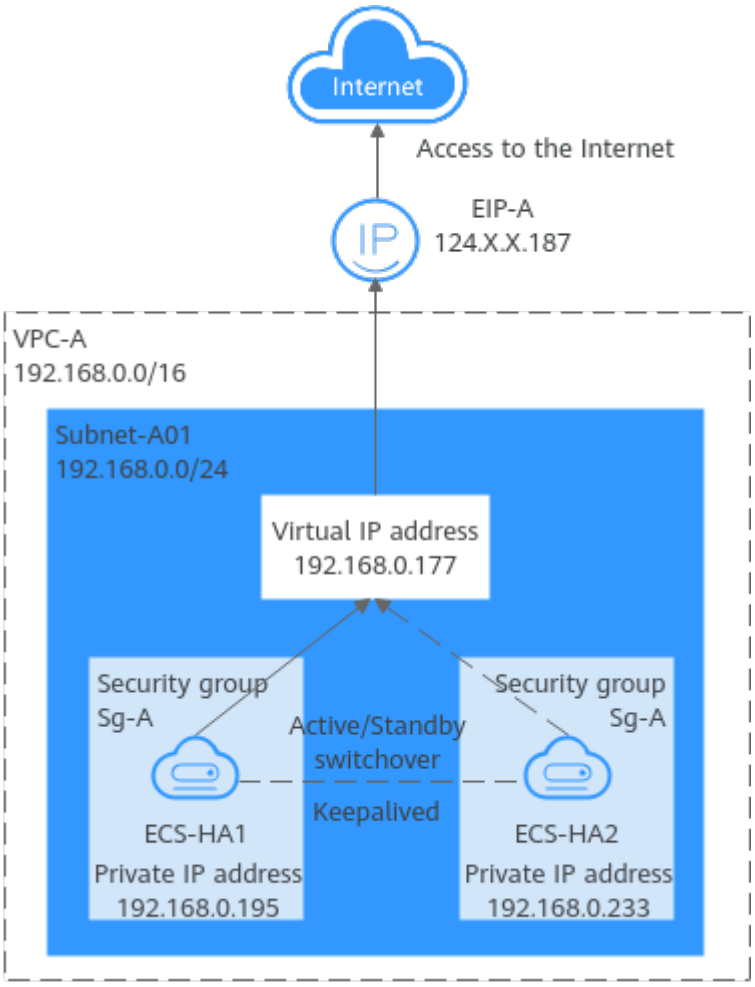
Figure 4-2 shows a high-availability cluster that is set up using a virtual IP address and Keepalived. They work as follows:

1. Virtual IP address **192.168.0.177** is bound to **ECS-HA1** and **ECS-HA2**. Keepalived is configured on the two ECSs.
2. EIP **EIP-A** is bound to the virtual IP address so that the ECSs can be accessed from the Internet.

In this cluster, **ECS-HA1** works as the active ECS and provides services accessible from the Internet using **EIP-A**. **ECS-HA2** works as the standby ECS, with no services deployed on it. If **ECS-HA1** goes down, **ECS-HA2** takes over services, ensuring service continuity.

For details about how to set up an HA cluster, see **Using a Virtual IP Address and Keepalived to Set Up a High-Availability Web Cluster**.

Figure 4-2 A high-availability cluster using a virtual IP address and Keepalived



Using a Virtual IP Address and Keepalived/LVS to Set Up a High-Availability Load Balancing Cluster

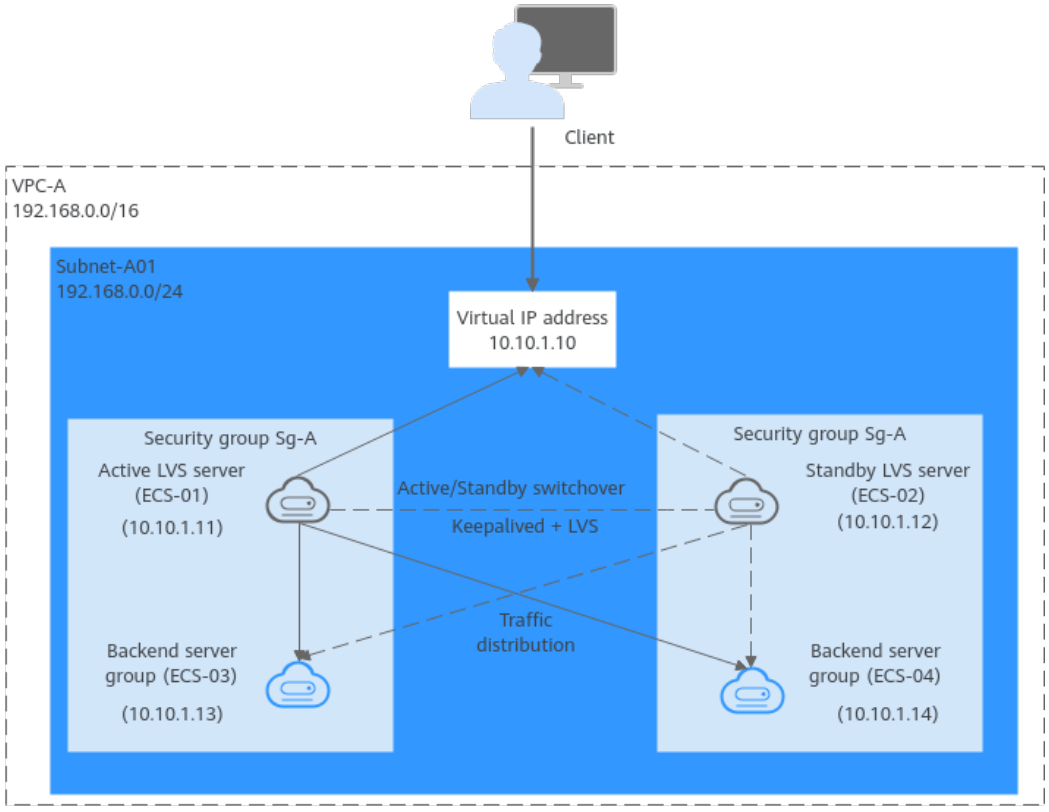
As shown in **Figure 4-3**, a virtual IP address, Keepalived, and LVS are used to set up an HA load balancing cluster. LVS is used for load balancing, and Keepalived is used for high availability. They work as follows:

1. Virtual IP address **10.10.1.10** is bound to **ECS-01** and **ECS-02**. Keepalived and LVS (DR mode) are configured on **ECS-01** and **ECS-02** to set up the active/standby LVS servers. In this way, requests from clients can be evenly distributed to different backend servers.
2. **ECS-03** and **ECS-04** are configured as backend servers to handle service requests.
3. The source/destination check option needs to be disabled.

   When you bind a virtual IP address to an ECS, the source/destination check option of the ECS's network interface is automatically disabled. If the option is not disabled, disable it.

In this load balancing cluster, **ECS-01** works as the active LVS server to distribute requests from clients. If **ECS-01** is faulty, **ECS-02** takes over and distributes requests from clients, ensuring high availability of the LVS cluster.

Figure 4-3 A high-availability cluster using a virtual IP address and Keepalived/LVS



NOTE

For details about how to install and configure Keepalived and LVS services and how to configure backend servers, see the common practices in the industry.

Virtual IP Address Quotas

Table 4-1 lists the quotas about virtual IP addresses. Some default quotas can be increased.

Table 4-1 Virtual IP address quotas

Item	Default Quota	Adjustable
Maximum number of virtual IP addresses per region	2	Yes. For details, see **Managing Quotas**.
Maximum number of EIPs that a virtual IP address can be bound to	1	No
Maximum number of instances (including cloud servers and network interfaces) that a virtual IP address can be bound to	10	No

Notes and Constraints

- If a cloud server has multiple network interfaces that are in the same subnet, you are not advised to bind virtual IP addresses to the network interfaces. Using the virtual IP addresses may cause route conflicts on the server, which would lead to communication failures.
- A virtual IP address is assigned from a VPC subnet. They can only be bound to a cloud server in the same subnet as the virtual IP address.
- Virtual IP addresses and extended network interfaces cannot be used to directly access Huawei Cloud services, such as DNS. You can use VPCEP to access these services. For details, see **Buying a VPC Endpoint**.

4.2 Assigning a Virtual IP Address

Scenarios

A virtual IP address is an IP address assigned from a VPC subnet. It can be assigned and released independently. You can follow the instructions in this section to assign a virtual IP address.

Procedure

1. Log in to the management console.
2. Click  in the upper left corner and select the desired region and project.
3. Click  in the upper left corner and choose **Networking** > **Virtual Private Cloud**.

   The **Virtual Private Cloud** page is displayed.
4. In the navigation pane on the left, choose **Virtual Private Cloud** > **Subnets**.
5. In the subnet list, click the name of the subnet where a virtual IP address is to be assigned.

   The subnet details page is displayed.
6. Switch to the **IP Addresses** tab and click **Assign Virtual IP Address**.

   The **Assign Virtual IP Address** page is displayed.
7. Set the parameters as required based on the below table.

Table 4-2 Virtual IP address parameters

Parameter	Description	Example Value
Subnet	Subnet from which a virtual IP address will be assigned. The current subnet is selected by default.	Subnet-A01

Parameter	Description	Example Value
IP Address Version	IP address version. This parameter is only shown when IPv6 is enabled for the subnet. There are two options: • IPv4 • IPv6	IPv4
Assignment Mode	How virtual IP addresses are assigned. There are two options: • **Automatic**: The system assigns a virtual IP address from the subnet. • **Manual**: You can specify a virtual IP address.	Manual
IP Address	Virtual IP addresses. This parameter is required if **Assignment Mode** is set to **Manual**. Specify an available IP address from the subnet CIDR block.	192.168.0.15

8. After setting the parameters, click **OK**.

   You can then check the assigned virtual IP address in the virtual IP address list.

4.3 Binding a Virtual IP Address to an Instance or EIP

Scenarios

You can bind a virtual IP address to an instance or EIP. An instance can be a cloud server, a network interface, or a Layer 2 connection.

- Bind a virtual IP address to an instance. You can:
  - Bind one or more virtual IP addresses to an instance.
  - Bind a virtual IP address to multiple instances.
- Bind a virtual IP address to an EIP to enable public network communication.

Constraints

It is recommended that a maximum of eight virtual IP addresses be bound to an ECS. If an ECS has multiple virtual IP addresses, each virtual IP address is used by a specific service. If there are too many services, the ECS may become overloaded and compromise user experience.

Binding a Virtual IP Address to an EIP or ECS on the Console

1. Log in to the management console.

2. Click in the upper left corner and select the desired region and project.
3. Click in the upper left corner and choose **Networking** > **Virtual Private Cloud**.

   The **Virtual Private Cloud** page is displayed.
4. In the navigation pane on the left, choose **Virtual Private Cloud** > **Subnets**.

   The **Subnets** page is displayed.
5. In the subnet list, locate the target subnet and click its name.

   The subnet details page is displayed.
6. On the **IP Addresses** tab, bind an EIP to the virtual IP address:
   a. Locate the row that contains the virtual IP address and click **Bind to EIP** in the **Operation** column.

      The **Bind to EIP** dialog box is displayed.
   b. Select an EIP and click **OK**.

      In the virtual IP address list, you can view the bound EIP.
7. On the **IP Addresses** tab, bind an instance to the virtual IP address:
   a. Locate the row that contains the virtual IP address and click **Bind to Instance** in the **Operation** column.

      The **Bind to Instance** dialog box is displayed.
   b. Select an instance and click **OK**.

      In the virtual IP address list, you can view the bound instance.

NOTICE

- After you bind one or more virtual IP addresses to an ECS, you need to manually configure the virtual IP addresses on the ECS. For details, see **Configuring a Virtual IP Address for an ECS**.
- If you want to bind a virtual IP address to multiple ECSs and use Keepalived to build an HA cluster, see **Using a Virtual IP Address and Keepalived to Set Up a High-Availability Web Cluster**.

Configuring a Virtual IP Address for an ECS

After you bind one or more virtual IP addresses to an ECS on the console, you must log in to the ECS to manually configure these virtual IP address.

The following OSs are used as examples here. For other OSs, see the help documents on their official websites.

- Linux: CentOS 7.2 64bit and Ubuntu 22.04 server 64bit
- Windows: Windows Server

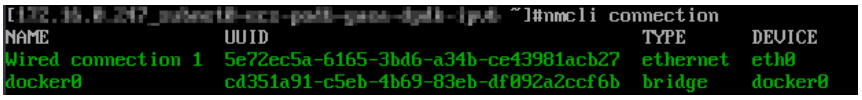
Linux (CentOS)

The following uses CentOS 7.2 64bit as an example.

1. Obtain the network interface that the virtual IP address is to be bound and the connection of the network interface:

nmcli connection

Information similar to the following is displayed:



The command output in this example is described as follows:

- **eth0** in the **DEVICE** column indicates the network interface that the virtual IP address is to be bound.
- **Wired connection 1** in the **NAME** column indicates the connection of the network interface.

2. Add the virtual IP address for the connection:

   nmcli connection modify "*<connection-name-of-the-network-interface>***" +ipv4.addresses** *<virtual-IP-address>*

   Configure the parameters as follows:

   - *connection-name-of-the-network-interface*: The connection name of the network interface obtained in **1**. In this example, the connection name is **Wired connection 1**.
   - *virtual-IP-address*: Enter the virtual IP address to be added. If you add multiple virtual IP addresses at a time, separate every two with a comma (,).

   Example commands:

   - Adding a single virtual IP address: **nmcli connection modify "Wired connection 1" +ipv4.addresses 172.16.0.125**
   - Adding multiple virtual IP addresses: **nmcli connection modify "Wired connection 1" +ipv4.addresses 172.16.0.125,172.16.0.126**

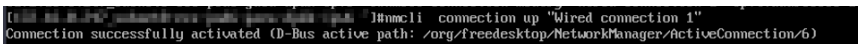
3. Make the configuration in **2** take effect:

   nmcli connection up "*<connection-name-of-the-network-interface>***"**

   In this example, run the following command:

   nmcli connection up "Wired connection 1"

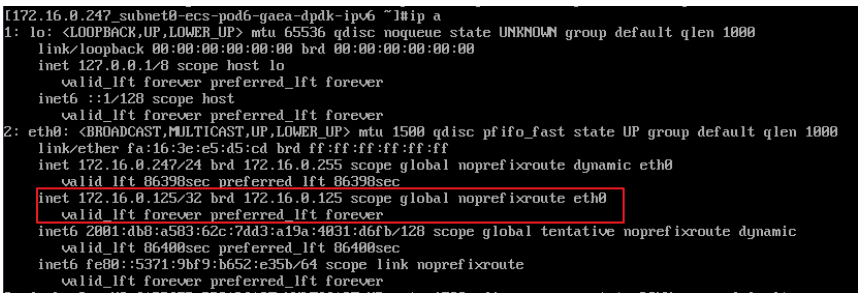
   Information similar to the following is displayed:



4. Check whether the virtual IP address has been bound:

   ip a

   Information similar to the following is displayed. In the command output, the virtual IP address 172.16.0.125, is bound to network interface eth0.



NOTE

To delete an added virtual IP address, perform the following steps:

1. Delete the virtual IP address from the connection of the network interface:

   nmcli connection modify "*<connection-name-of-the-network-interface>***" -ipv4.addresses** *<virtual-IP-address>*

   To delete multiple virtual IP addresses at a time, separate every two with a comma (,). Example commands are as follows:

   - Deleting a single virtual IP address: **nmcli connection modify "Wired connection 1" -ipv4.addresses 172.16.0.125**
   - Deleting multiple virtual IP addresses: **nmcli connection modify "Wired connection 1" -ipv4.addresses 172.16.0.125,172.16.0.126**

2. Make the deletion take effect by referring to **3**.

Linux (Ubuntu)

The following uses Ubuntu 22.04 server 64bit as an example. If the ECS runs **Ubuntu 22** or **Ubuntu 20**, perform the following operations:

1. Obtain the network interface that the virtual IP address is to be bound:

   ifconfig

   Information similar to the following is displayed. In this example, the network interface bound to the virtual IP address is **eth0**.

```
root@ecs-X-ubantu:~# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 172.16.0.210  netmask 255.255.255.0  broadcast 172.16.0.255
        inet6 fe80::f816:3eff:fe01:f1c3  prefixlen 64  scopeid 0x20<link>
        ether fa:16:3e:01:f1:c3  txqueuelen 1000  (Ethernet)
        RX packets 43915  bytes 63606486 (63.6 MB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 3364  bytes 455617 (455.6 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
...
```

2. Switch to the **/etc/netplan** directory:

   cd /etc/netplan

3. Add a virtual IP address to the network interface.

   a. Open the configuration file **01-netcfg.yaml**:

      vim 01-netcfg.yaml

   b. Press **i** to enter the editing mode.

   c. In the network interface configuration area, add a virtual IP address.

      In this example, add a virtual IP address for **eth0**:

      addresses:

      - 172.16.0.26/32

      The file content is as follows:

```
network:
  version: 2
  renderer: NetworkManager
  ethernets:
    eth0:
      dhcp4: true
      addresses:
      - 172.16.0.26/32
    eth1:
      dhcp4: true
```



```
    eth2:
      dhcp4: true
    eth3:
      dhcp4: true
    eth4:
      dhcp4: true
```

d. Press **Esc**, enter **:wq!**, save the configuration, and exit.

4. Make the configuration in **3** take effect:

   netplan apply

5. Check whether the virtual IP address has been bound:

   ip a

   Information similar to the following is displayed. In the command output, virtual IP address 172.16.0.26 is bound to network interface eth0.

```
root@ecs-X-ubantu:/etc/netplan# ip a
...
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default
qlen 1000
    link/ether fa:16:3e:01:f1:c3 brd ff:ff:ff:ff:ff:ff
    altname enp0s3
    altname ens3
    inet 172.16.0.26/32 scope global noprefixroute eth0
       valid_lft forever preferred_lft forever
    inet 172.16.0.210/24 brd 172.16.0.255 scope global dynamic noprefixroute eth0
       valid_lft 107999971sec preferred_lft 107999971sec
    inet6 fe80::f816:3eff:fe01:f1c3/64 scope link
       valid_lft forever preferred_lft forever
```

NOTE

To delete an added virtual IP address, perform the following steps:

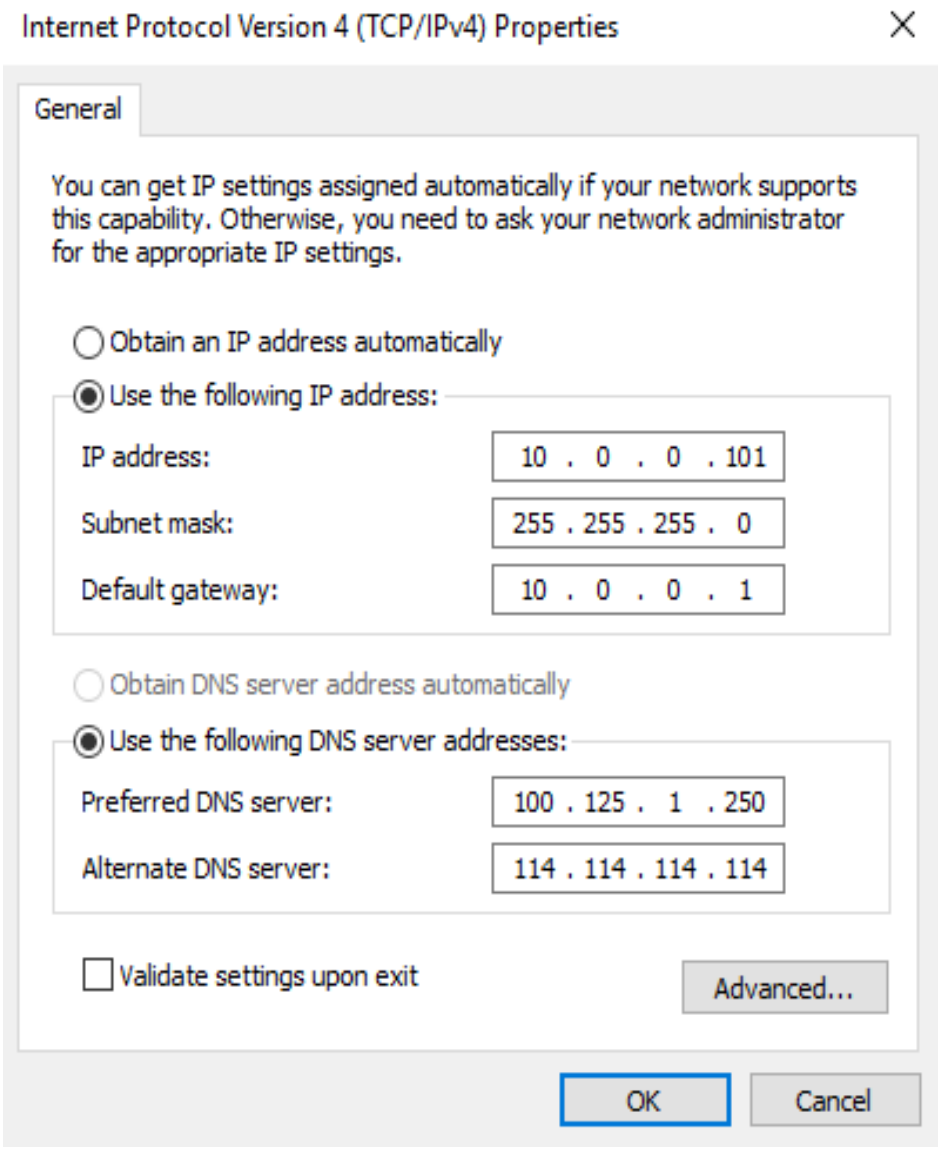
1. Open the configuration file **01-netcfg.yaml** and delete the virtual IP address of the corresponding network interface by referring to **3**.
2. Make the deletion take effect by referring to **4**.

Windows OS

The following operations use Windows Server as an example.

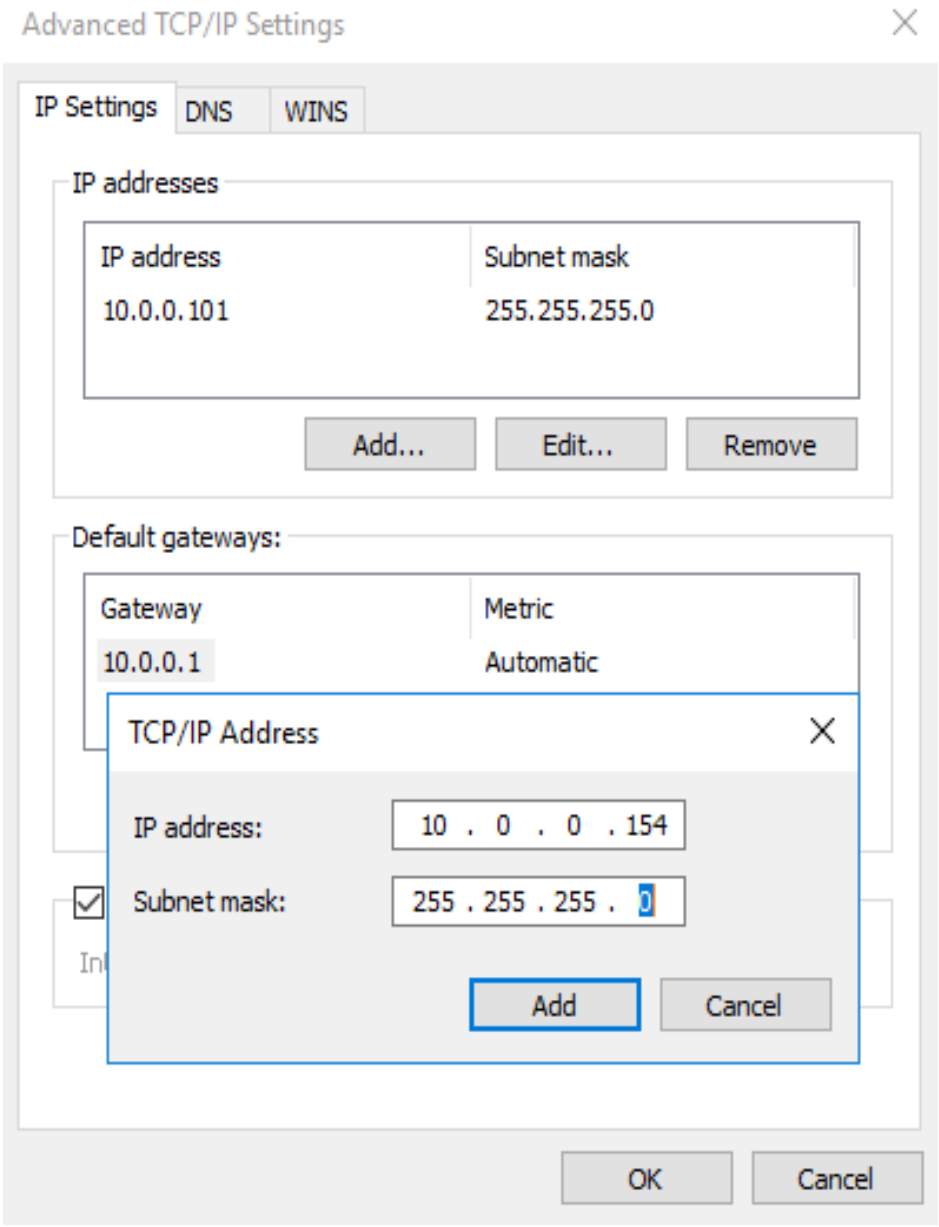
1. In **Control Panel**, click **Network and Sharing Center**, and click the corresponding local connection.
2. On the displayed page, click **Properties**.
3. On the **Network** tab page, select **Internet Protocol Version 4 (TCP/IPv4)**.
4. Click **Properties**.
5. Select **Use the following IP address** and set **IP address** to the private IP address of the ECS, for example, 10.0.0.101.

Figure 4-4 Configuring private IP address



6. Click **Advanced**.
7. On the **IP Settings** tab, click **Add** in the **IP addresses** area.
   Add the virtual IP address, for example, 10.0.0.154.

Figure 4-5 Configuring virtual IP address



8. Click **OK**.
9. In the **Start** menu, open the Windows command line window and run the following command to check whether the virtual IP address has been configured:

   ipconfig /all

   In the command output, **IPv4 Address** is the virtual IP address 10.0.0.154, indicating that the virtual IP address of the ECS's network interface has been correctly configured.

Helpful Links

- **Why Can't the Virtual IP Address Be Pinged After It Is Bound to an ECS Network Interface?**

- **What Are the Differences Between EIPs, Private IP Addresses, and Virtual IP Addresses?**

4.4 Unbinding a Virtual IP Address from an Instance or EIP

Scenarios

You can unbind a virtual IP address from an instance or EIP. An instance can be a cloud server, a network interface, or a Layer 2 connection. For detailed operations, see:

- **Unbinding a Virtual IP Address from an Instance**
- **Unbinding a Virtual IP Address from an EIP**

Unbinding a Virtual IP Address from an Instance

1. Log in to the management console.
2. Click in the upper left corner and select the desired region and project.
3. Click in the upper left corner and choose **Networking** > **Virtual Private Cloud**.

   The **Virtual Private Cloud** page is displayed.
4. In the navigation pane on the left, choose **Virtual Private Cloud** > **Subnets**.

   The **Subnets** page is displayed.
5. In the subnet list, locate the target subnet and click its name.

   The subnet details page is displayed.
6. Click the **IP Addresses** tab.

   The virtual IP address list is displayed.
7. Locate the row that contains the virtual IP address, click **More** in the **Operation** column, and select **Unbind from Instance**.

   A confirmation dialog box is displayed.
8. In the displayed dialog box, perform the following operations to unbind the virtual IP address from the instance:
   a. Select the type of the instance bound to the virtual IP address.
   b. Locate the row that contains the instance and click **Unbind** in the **Operation** column.

      A confirmation dialog box is displayed.
   c. Confirm the information and click **OK**.

Unbinding a Virtual IP Address from an EIP

1. Log in to the management console.
2. Click in the upper left corner and select the desired region and project.
3. Click in the upper left corner and choose **Networking** > **Virtual Private Cloud**.

The **Virtual Private Cloud** page is displayed.

4. In the navigation pane on the left, choose **Virtual Private Cloud** > **Subnets**.

   The **Subnets** page is displayed.

5. In the subnet list, locate the target subnet and click its name.

   The subnet details page is displayed.

6. Click the **IP Addresses** tab.

   The virtual IP address list is displayed.

7. Locate the target virtual IP address, click **More** in the **Operation** column, and select **Unbind from EIP**.

   A confirmation dialog box is displayed.

8. Confirm the information and click **OK**.

4.5 Releasing a Virtual IP Address

Scenarios

If you no longer need a virtual IP address or a reserved virtual IP address, you can release it to avoid wasting resources.

Constraints

If you want to release a virtual IP address that is being used by a resource, refer to **Table 4-3**.

Table 4-3 Releasing a virtual IP address that is being used by a resource

Prompts	Cause Analysis and Solution
Scenario 1: This virtual IP address is bound to a resource. Unbind the virtual IP address first.	This virtual IP address is being used by cloud resources such as an EIP or an ECS. For details, see **Unbinding a Virtual IP Address from an Instance or EIP**. Release the virtual IP address.
Scenario 2: This operation cannot be performed because the IP address is being used by a system component.	The virtual IP address is being used by an instance. Delete the instance, which will also release the virtual IP address. Search for the instance based on the instance information displayed on the virtual IP address console and delete the instance. • RDS DB instance: **RDS Documentation** • CCE instance: **CCE Documentation** • API gateway: **API Gateway Documentation**

Figure 4-6 Scenario 1: Virtual IP address cannot be released

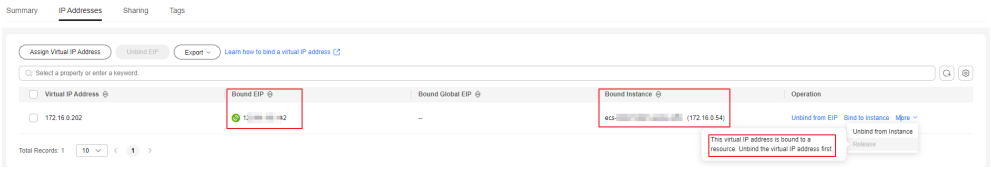
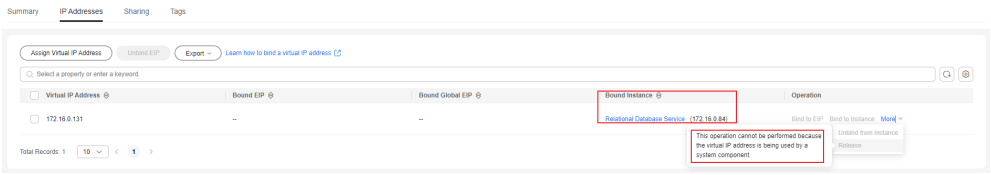


Figure 4-7 Scenario 2: Virtual IP address cannot be released



Procedure

1. Log in to the management console.
2. Click  in the upper left corner and select the desired region and project.
3. Click  in the upper left corner and choose **Networking** > **Virtual Private Cloud**.

   The **Virtual Private Cloud** page is displayed.
4. In the navigation pane on the left, choose **Virtual Private Cloud** > **Subnets**.
5. Click the name of the subnet that the virtual IP address belongs to.
6. Click the **IP Addresses** tab, locate the row that contains the virtual IP address to be released, click **More** in the **Operation** column, and select **Release**.

   A confirmation dialog box is displayed.
7. Confirm the information and click **OK**.

4.6 Virtual IP Address Configuration Example

4.6.1 Using a Virtual IP Address and Keepalived to Set Up a High-Availability Web Cluster

Scenarios

A virtual IP address is a private IP address assigned from a VPC subnet. You can use a virtual IP address and Keepalived to set up a high-availability active/standby web cluster. In such a cluster, if the active ECS goes down, the virtual IP address is bound to the standby ECS to provide services. This section describes how to use a virtual IP address and Keepalived to set up a high-availability web cluster.

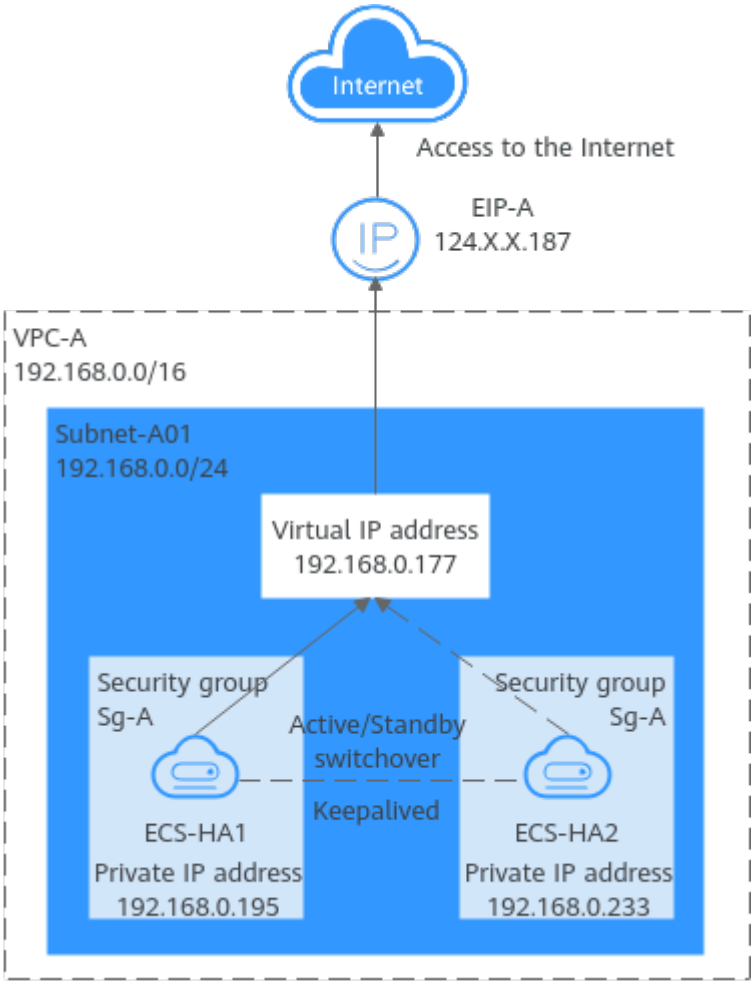
Architecture

Figure 4-8 shows a high-availability web cluster using Keepalived. In this architecture, virtual IP address **192.168.0.177** is bound to **ECS-HA1** and **ECS-HA2**.

To allow **ECS-HA1** and **ECS-HA2** to access and be accessed from the Internet, an EIP (**EIP-A**) is bound to the virtual IP address. They work as follows:

1. **ECS-HA1** works as the active ECS and provides services accessible from the Internet using **EIP-A**. **ECS-HA2** works as the standby ECS, with no services deployed on it.
2. If **ECS-HA1** goes down, **ECS-HA2** takes over services, ensuring service continuity.

Figure 4-8 A high-availability web cluster using a virtual IP address and Keepalived



Advantages

A high-availability cluster can have one active ECS and one standby ECS or one active ECS and multiple standby ECSs. You can bind a virtual IP address to these ECSs. If the active ECS goes down, the standby ECS becomes the active ECS and continues to provide services.

Notes and Constraints

All servers of the HA cluster must be in the same subnet.

Resource Planning

In this example, the VPC, subnet, virtual IP address, EIP, and ECSs must be in the same region but can be in different AZs.

NOTE

The following resource details are only for your reference. You can modify them if needed.

Table 4-4 Resource planning

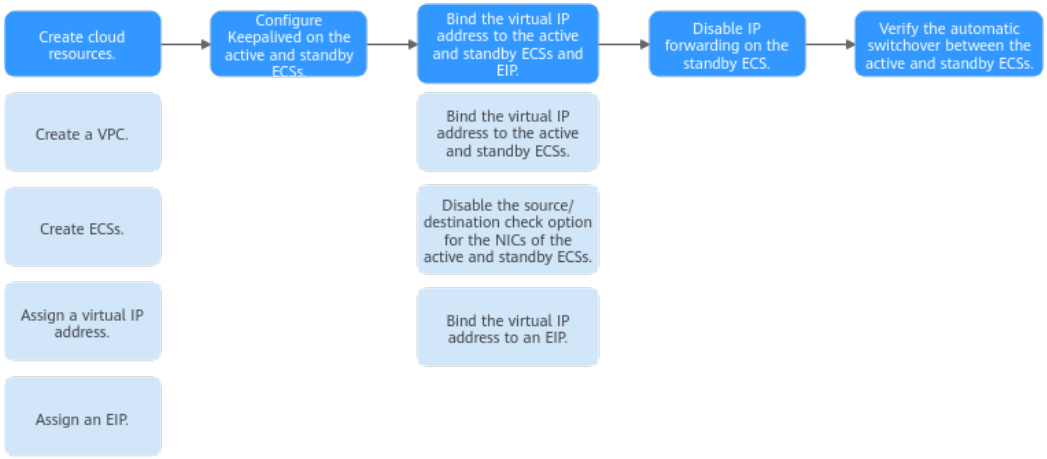
Resource Type	Quantity	Description
VPC and subnet	1	• VPC name: Set it as needed. In this example, **VPC-A** is used. • VPC IPv4 CIDR block: Set it as needed. In this example, **192.168.0.0/16** is used. • Subnet name: Set it as needed. In this example, **Subnet-A01** is used. • Subnet IPv4 CIDR block: Set it as needed. In this example, **192.168.0.0/24** is used.
ECS	2	In this example, two ECSs are required for active/standby switchover. Configure the two ECSs as follows: • **Name**: Set this parameter as needed. In this example, the two ECSs are named **ECS-HA1** and **ECS-HA2**. • **Image**: Select an image as needed. In this example, a public image (CentOS 7.8 64bit) is used. • **System Disk**: **General Purpose SSD** | 40 GiB • **Data Disk**: In this example, no data disk is required. You can attach data disks based on service requirements and ensure data consistency between the two ECSs. • Network parameters – VPC: Select a VPC. In this example, **VPC-A** is used. – Subnet: Select a subnet. In this example, **Subnet-A01** is used. • **Security Group**: Select a security group as needed. In this example, **ECS-HA1** and **ECS-HA2** are associated with the same security group (**Sg-A**). • Private IP address: Specify **192.168.0.195** for **ECS-HA1** and **192.168.0.233** for **ECS-HA2**.

Resource Type	Quantity	Description
Virtual IP address	1	Assign a virtual IP address from **Subnet-A01**. • **Assignment Mode**: Set it as needed. In this example, **Automatic** is selected. • Virtual IP address: **192.168.0.177** is used in this example. • Instances: Bind **192.168.0.177** to **ECS-HA1** and **ECS-HA2**. • EIP: Bind **192.168.0.177** to **EIP-A**.
EIP	1	• **Billing Mode:** Select a billing mode as needed. In this example, **Pay-per-use** is used. • **EIP Name**: Set it as needed. In this example, **EIP-A** is used. • **EIP**: The IP address is randomly assigned. In this example, **124.X.X.187** is used.

Procedure

You can follow the process in **Figure 4-9** to set up a high-availability web cluster using a virtual IP address and Keepalived

Figure 4-9 Process for setting up a high-availability web cluster



Step 1: Create Cloud Resources

1. Create a VPC and subnet.
   For details, see **Creating a VPC and Subnet**.
2. Create two ECSs, one as the active ECS and the other as the standby ECS.
   For details, see **Purchasing a Custom ECS**.
   Configure the ECSs as follows:

- **Network**: Select **VPC-A** and **Subnet-A01** you have created.
- **Security Group**: Create security group **Sg-A** and add inbound and outbound rules to it. Each security group comes with preset rules. You need to check and modify the rules as required.

  Add rules in **Table 4-5** to **Sg-A** and associate **Sg-A** with **ECS-HA1** and **ECS-HA2**.

Table 4-5 Sg-A rules

Direction	Action	Type	Protocol & Port	Source/ Destination	Description
Inbound	Allow	IPv4	TCP: 22	Source: 0.0.0.0/0	Allows remote logins to Linux ECSs over SSH port 22.
Inbound	Allow	IPv4	TCP: 3389	Source: 0.0.0.0/0	Allows remote logins to Windows ECSs over RDP port 3389.
Inbound	Allow	IPv4	TCP: 80	Source: 0.0.0.0/0	Allows external access to the website deployed on the ECSs over HTTP port 80.
Inbound	Allow	IPv4	All	Source: current security group (**Sg-A**)	Allows the ECSs in **Sg-A** to communicate with each other using IPv4 addresses.
Inbound	Allow	IPv6	All	Source: current security group (**Sg-A**)	Allows the ECSs in **sg-A** to communicate with each other using IPv6 addresses.
Outbound	Allow	IPv4	All	Destination: 0.0.0.0/0	Allows ECSs in **Sg-A** to access the Internet using IPv4 addresses.
Outbound	Allow	IPv6	All	Destination: ::/0	Allows ECSs in **Sg-A** to access the Internet using IPv6 addresses.

NOTICE

In this example, **Source** is set to **0.0.0.0/0**, which allows any external IP address to remotely log in to ECSs in **Sg-A**. To ensure security, you are advised to set **Source** to a specific IP address, for example, the IP address of your local PC.

If your ECSs are associated with different security groups, you need to add rules in **Table 4-6** to allow the ECSs in the two security groups to communicate with each other.

Table 4-6 Rules of security groups **Sg-A** and **Sg-B**

Security Group	Direction	Action	Type	Protocol & Port	Source/ Destination	Description
Sg-A	Inbound	Allow	IPv4	All	Source: Sg-B	Allows ECSs in **Sg-B** to access those in **Sg-A** over any IPv4 protocol and port.
Sg-B	Inbound	Allow	IPv4	All	Source: Sg-A	Allows ECSs in **Sg-A** to access those in **Sg-B** over any IPv4 protocol and port.

- **EIP**: Select **Not required**.

3. Assign a virtual IP address from **Subnet-A01**.
   For details, see **Assigning a Virtual IP Address**.
4. Assign an EIP.
   For details, see **Assigning an EIP**.

Step 2: Configure Keepalived on ECS-HA1 and ECS-HA2

1. Configure Keepalived on **ECS-HA1**.
   a. Bind **EIP-A** (**124.X.X.187**) to **ECS-HA1**.
      For details, see **Binding an EIP to an ECS**.
   b. Remotely log in to **ECS-HA1**.
      For details, see **How Do I Log In to My ECS?**
   c. Run the following command to install the Nginx and Keepalived packages and related dependency packages:
      yum install nginx keepalived -y
      If information similar to the following is displayed, the installation is complete:

```
[root@ecs-ha1 ~]# yum install nginx keepalived -y
Loaded plugins: fastestmirror
Determining fastest mirrors
base
                        | 3.6 kB  00:00:00
epel
                        | 4.3 kB  00:00:00
extras
                        | 2.9 kB  00:00:00
updates
                          | 2.9 kB  00:00:00
(1/7): epel/x86_64/
group
        | 399 kB  00:00:00
```



```
(2/7): epel/x86_64/
updateinfo
          | 1.0 MB  00:00:00
(3/7): base/7/x86_64/
primary_db
         | 6.1 MB  00:00:00
(4/7): base/7/x86_64/
group_gz
        | 153 kB  00:00:00
(5/7): epel/x86_64/
primary_db
           | 8.7 MB  00:00:00
(6/7): extras/7/x86_64/
primary_db
        | 253 kB  00:00:00
(7/7): updates/7/x86_64/primary_db

.....
Dependency Installed:
  centos-indexhtml.noarch 0:7-9.el7.centos                    gperftools-libs.x86_64
0:2.6.1-1.el7                lm_sensors-libs.x86_64 0:3.4.0-8.20160601gitf9185e5.el7_9.1
  net-snmp-agent-libs.x86_64 1:5.7.2-49.el7_9.4               net-snmp-libs.x86_64
1:5.7.2-49.el7_9.4              nginx-filesystem.noarch 1:1.20.1-10.el7
  openssl11-libs.x86_64 1:1.1.1k-7.el7

Complete!
```

d. Modify the Nginx configuration file.

   i. Run the following command to open the **/etc/nginx/nginx.conf** file:

      vim /etc/nginx/nginx.conf

   ii. Press **i** to enter the editing mode.

   iii. Replace the original content with the following:

```
user root;
worker_processes 1;
#error_log logs/error.log;
#error_log logs/error.log notice;
#error_log logs/error.log info;
#pid logs/nginx.pid;
events {
      worker_connections 1024;
      }
http {
    include mime.types;
    default_type application/octet-stream;
    #log_format main '$remote_addr  - $remote_user [$time_local] "$request" '
    # '$status $body_bytes_sent  "$http_referer" '
    # '"$http_user_agent"  "$http_x_forwarded_for"';
    #access_log logs/access.log main;
    sendfile on;
    #tcp_nopush on;
    #keepalive_timeout 0;
    keepalive_timeout 65;
    #gzip on;
    server {
        listen 80;
        server_name localhost;
        #charset koi8-r;
        #access_log logs/host.access.log main;
        location / {
                root html;
                index index.html index.htm;
                }
        #error_page 404  /404.html;
        # redirect server error pages to the static page /50x.html
        error_page 500 502 503 504 /50x.html;
        location =  /50x.html {
                        root html;
```



```
                }
        }
}
```

iv. Press **ESC** to exit and enter **:wq!** to save the configuration.

e. Modify the **index.html** file to verify whether the website is successfully accessed.

i. Run the following command to open the **/usr/share/nginx/html/index.html** file:

vim /usr/share/nginx/html/index.html

ii. Press **i** to enter the editing mode.

iii. Replace the original content with the following:

```
Welcome to ECS-HA1
```

iv. Press **ESC** to exit and enter **:wq!** to save the configuration.

f. Run the following commands to set the automatic startup of Nginx upon ECS startup:

systemctl enable nginx

systemctl start nginx.service

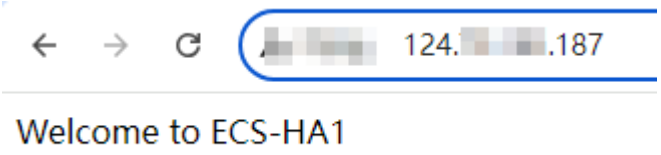
Information similar to the following is displayed:

```
[root@ecs-ha1 ~]# systemctl enable nginx
Created symlink from /etc/systemd/system/multi-user.target.wants/nginx.service to /usr/lib/systemd/system/nginx.service.
[root@ecs-ha1 ~]# systemctl start nginx.service
```

g. Open a browser, enter the EIP address (**124.X.X.187**), and press **Enter** to verify the access to a single Nginx node.

If the web page shown in the following figure is displayed, Nginx is successfully configured for **ECS-HA1**.

Figure 4-10 ECS-HA1 accessed



h. Modify the Keepalived configuration file.

i. Run the following command to open the **/etc/keepalived/keepalived.conf** file:

vim /etc/keepalived/keepalived.conf

ii. Press **i** to enter the editing mode.

iii. Replace the IP parameters in the configuration file as follows:

- **mcast_src_ip** and **unicast_src_ip**: Change their values to the private IP address of an ECS. In this example, private IP address **192.168.0.195** of **ECS-HA1** is used.
- **virtual_ipaddress**: Change the value to a virtual IP address. In this example, **192.168.0.177** is used.

```
! Configuration File for keepalived
global_defs {
router_id master-node
}
```



```
vrrp_script chk_http_port {
        script  "/etc/keepalived/chk_nginx.sh"
        interval 2
        weight -5
        fall 2
        rise 1
      }
vrrp_instance VI_1 {
    state BACKUP
    interface eth0
    mcast_src_ip 192.168.0.195
    virtual_router_id 51
    priority 100
    advert_int 1
    authentication {
            auth_type PASS
            auth_pass 1111
            }
    unicast_src_ip 192.168.0.195
    virtual_ipaddress {
                192.168.0.177
              }
track_script {
    chk_http_port
    }
}
```

iv. Press **ESC** to exit and enter **:wq!** to save the configuration.

i. Configure the Nginx monitoring script.

i. Run the following command to open the **/etc/keepalived/chk_nginx.sh** file:

vim /etc/keepalived/chk_nginx.sh

ii. Press **i** to enter the editing mode.

iii. Replace the original content with the following:

```
#!/bin/bash
counter=$(ps -C nginx --no-heading|wc -l)
if [ "${counter}" = "0"  ]; then
    systemctl start nginx.service
    sleep 2
    counter=$(ps -C nginx  --no-heading|wc -l)
    if [ "${counter}" =  "0" ]; then
        systemctl stop keepalived.service
    fi
fi
```

iv. Press **ESC** to exit and enter **:wq!** to save the configuration.

j. Run the following command to assign execute permissions to the **chk_nginx.sh** file:

chmod +x /etc/keepalived/chk_nginx.sh

k. Run the following commands to set the automatic startup of Keepalived upon ECS startup:

systemctl enable keepalived

systemctl start keepalived.service

l. Unbind **EIP-A** from **ECS-HA1**.

For details, see **Unbinding an EIP**.

2. Configure Keepalived on **ECS-HA2**.

a. Bind **EIP-A** (**124.X.X.187**) to **ECS-HA2**.

For details, see **Binding an EIP to an ECS**.

b. Remotely log in to **ECS-HA2**.

For details, see **How Do I Log In to My ECS?**

c. Run the following command to install the Nginx and Keepalived packages and related dependency packages:

yum install nginx keepalived -y

If information similar to the following is displayed, the installation is complete:

```
[root@ecs-ha2 ~]# yum install nginx keepalived -y
Loaded plugins: fastestmirror
Determining fastest mirrors
base
                          | 3.6 kB  00:00:00
epel
                          | 4.3 kB  00:00:00
extras
                          | 2.9 kB  00:00:00
updates
                            | 2.9 kB  00:00:00
(1/7): epel/x86_64/
group
         | 399 kB  00:00:00
(2/7): epel/x86_64/
updateinfo
           | 1.0 MB  00:00:00
(3/7): base/7/x86_64/
primary_db
          | 6.1 MB  00:00:00
(4/7): base/7/x86_64/
group_gz
         | 153 kB  00:00:00
(5/7): epel/x86_64/
primary_db
            | 8.7 MB  00:00:00
(6/7): extras/7/x86_64/
primary_db
         | 253 kB  00:00:00
(7/7): updates/7/x86_64/primary_db

.....
Dependency Installed:
  centos-indexhtml.noarch 0:7-9.el7.centos                    gperftools-libs.x86_64
0:2.6.1-1.el7                  lm_sensors-libs.x86_64 0:3.4.0-8.20160601gitf9185e5.el7_9.1
  net-snmp-agent-libs.x86_64 1:5.7.2-49.el7_9.4               net-snmp-libs.x86_64
1:5.7.2-49.el7_9.4               nginx-filesystem.noarch 1:1.20.1-10.el7
  openssl11-libs.x86_64 1:1.1.1k-7.el7

Complete!
```

d. Modify the Nginx configuration file.

i. Run the following command to open the **/etc/nginx/nginx.conf** file:

vim /etc/nginx/nginx.conf

ii. Press **i** to enter the editing mode.

iii. Replace the original content with the following:

```
user root;
worker_processes 1;
#error_log logs/error.log;
#error_log logs/error.log notice;
#error_log logs/error.log info;
#pid logs/nginx.pid;
events {
    worker_connections 1024;
    }
http {
```



```
    include mime.types;
    default_type application/octet-stream;
    #log_format main '$remote_addr  - $remote_user [$time_local] "$request" '
    # '$status $body_bytes_sent  "$http_referer" '
    # '"$http_user_agent"  "$http_x_forwarded_for"';
    #access_log logs/access.log main;
    sendfile on;
    #tcp_nopush on;
    #keepalive_timeout 0;
    keepalive_timeout 65;
    #gzip on;
    server {
        listen 80;
        server_name localhost;
        #charset koi8-r;
        #access_log logs/host.access.log main;
        location / {
                root html;
                index index.html index.htm;
                }
        #error_page 404  /404.html;
        # redirect server error pages to the static page /50x.html
        error_page 500 502 503 504 /50x.html;
        location =  /50x.html {
                        root html;
                        }
        }
}
```

iv. Press **ESC** to exit and enter **:wq!** to save the configuration.

e. Modify the **index.html** file to verify whether the website is successfully accessed.

i. Run the following command to open the **/usr/share/nginx/html/index.html** file:

vim /usr/share/nginx/html/index.html

ii. Press **i** to enter the editing mode.

iii. Replace the original content with the following:

```
Welcome to ECS-HA2
```

iv. Press **ESC** to exit and enter **:wq!** to save the configuration.

f. Run the following commands to set the automatic startup of Nginx upon ECS startup:

systemctl enable nginx

systemctl start nginx.service

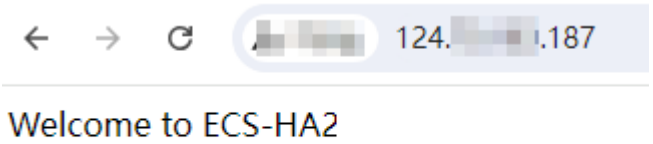
Information similar to the following is displayed:

```
[root@ecs-ha2 ~]# systemctl enable nginx
Created symlink from /etc/systemd/system/multi-user.target.wants/nginx.service to /usr/lib/systemd/system/nginx.service.
[root@ecs-ha2 ~]# systemctl start nginx.service
```

g. Open a browser, enter the EIP address (**124.X.X.187**), and press **Enter** to verify the access to a single Nginx node.

If the web page shown in the following figure is displayed, Nginx is successfully configured for **ECS-HA2**.

Figure 4-11 ECS-HA2 accessed



h. Modify the Keepalived configuration file.

i. Run the following command to open the **/etc/keepalived/keepalived.conf** file:

vim /etc/keepalived/keepalived.conf

ii. Press **i** to enter the editing mode.

iii. Replace the IP parameters in the configuration file as follows:

- **mcast_src_ip** and **unicast_src_ip**: Change their values to the private IP address of an ECS. In this example, private IP address of **ECS-HA2** (**192.168.0.233**) is used.
- **virtual_ipaddress**: Change the value to a virtual IP address. In this example, **192.168.0.177** is used.

```
! Configuration File for keepalived
global_defs {
router_id master-node
}
vrrp_script chk_http_port {
         script  "/etc/keepalived/chk_nginx.sh"
         interval 2
         weight -5
         fall 2
         rise 1
        }
vrrp_instance VI_1 {
    state BACKUP
    interface eth0
    mcast_src_ip 192.168.0.233
    virtual_router_id 51
    priority 100
    advert_int 1
    authentication {
            auth_type PASS
            auth_pass 1111
            }
    unicast_src_ip 192.168.0.233
    virtual_ipaddress {
                192.168.0.177
              }
track_script {
    chk_http_port
    }
}
```

iv. Press **ESC** to exit and enter **:wq!** to save the configuration.

i. Configure the Nginx monitoring script.

i. Run the following command to open the **/etc/keepalived/chk_nginx.sh** file:

vim /etc/keepalived/chk_nginx.sh

ii. Press **i** to enter the editing mode.

iii. Replace the original content with the following:

```
#!/bin/bash
counter=$(ps -C nginx --no-heading|wc -l)
if [ "${counter}" = "0"  ]; then
    systemctl start nginx.service
    sleep 2
    counter=$(ps -C nginx  --no-heading|wc -l)
    if [ "${counter}" =  "0" ]; then
        systemctl stop keepalived.service
    fi
fi
```

iv. Press **ESC** to exit and enter **:wq!** to save the configuration.

j. Run the following command to assign execute permissions to the **chk_nginx.sh** file:

chmod +x /etc/keepalived/chk_nginx.sh

k. Run the following commands to set the automatic startup of Keepalived upon ECS startup:

systemctl enable keepalived

systemctl start keepalived.service

l. Unbind **EIP-A** from **ECS-HA2**.

For details, see **Unbinding an EIP**.

Step 3: Bind the Virtual IP Address to the Active and Standby ECSs and EIP

1. Bind virtual IP address **192.168.0.177** to **ECS-HA1** and **ECS-HA2**.

   For details, see **Binding a Virtual IP Address to an Instance or EIP**.

2. Disable **Source/Destination Check** for the network interfaces of the active and standby ECSs.

   When you bind a virtual IP address to an ECS, **Source/Destination Check** is disabled by default. You can perform the following operations to check whether the function is disabled. If the function is not disabled, disable it.

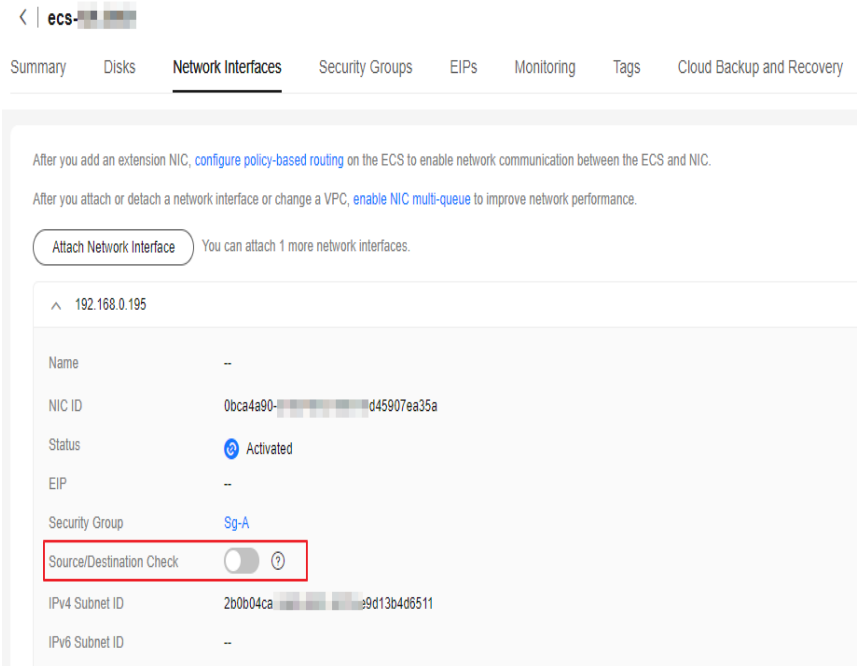
   a. In the ECS list, click the name of the target ECS.

      The ECS details page is displayed.

   b. On the **Network Interfaces** tab, click ⌄ to expand the details area and check whether **Source/Destination Check** is disabled.

      If the information shown in **Figure 4-12** is displayed, **Source/Destination Check** is disabled.

Figure 4-12 Disabling **Source/Destination Check**



3. Bind virtual IP address **192.168.0.177** to **EIP-A**.

   For details, see **Binding a Virtual IP Address to an Instance or EIP**.

Step 4: Disable IP Forwarding on the Standby ECS

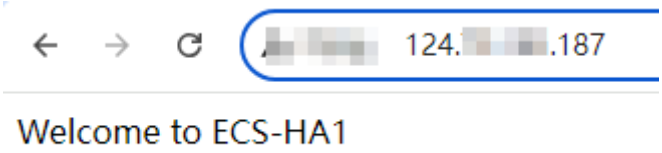
If a virtual IP address is bound to active/standby ECSs, you need to disable IP forwarding on the standby ECS. If an active/standby ECS switchover happens, ensure that IP forwarding of the new standby ECS is also disabled.

To make sure you do not miss any settings, it is better to disable IP forwarding on both of active and standby ECSs.

1. Open a browser, enter the EIP address (**124.X.X.187**), and press **Enter** to access the active ECS.

   If the following page is displayed, the **ECS-HA1** is used as the active ECS.

   Figure 4-13 The active ECS accessed



2. Remotely log in to the standby ECS (**ECS-HA2** in this example).

   For details, see **How Do I Log In to My ECS?**

3. Disable IP forwarding by following the operations in **Table 4-7**. In this example, the ECS runs the Linux OS.

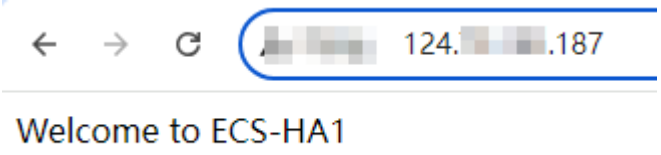
Table 4-7 Disabling IP forwarding

OS	Operations
Linux	1. Run the following command to switch to user **root**: **su root** 2. Run the following command to check whether IP forwarding is enabled: **cat /proc/sys/net/ipv4/ip_forward** In the command output, **1** indicates that IP forwarding is enabled, and **0** indicates that IP forwarding is disabled. The default value is **0**. • If **0** is displayed, no further action is required. • If **1** is displayed, go to the next step. 3. Use either of the following methods to modify the configuration file: Method 1 a. Run the following command to open the **/etc/sysctl.conf** file: **vim /etc/sysctl.conf** b. Press **i** to enter the editing mode. c. Set **net.ipv4.ip_forward** to **0**. d. Press **ESC** to exit and enter **:wq!** to save the configuration. Method 2 Run the **sed** command. An example command is as follows: **sed -i '/net.ipv4.ip_forward/s/1/0/g' /etc/sysctl.conf** 4. Run the following command to apply the modification: **sysctl -p /etc/sysctl.conf**
Windows	1. In the search box, enter **cmd** to open the **command prompt** window, and run the following command: **ipconfig/all** • In the command output, if the value of **IP Routing Enabled** is **No**, IP forwarding is disabled. • If **IP Routing Enabled** is **Yes**, IP forwarding is not disabled. Go to the next step. 2. Enter **regedit** in the search box to open the registry editor. 3. Set the value of **IPEnableRouter** under **HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Tcpip\Parameters** to **0**. • If the value is set to **0**, IP forwarding will be disabled. • If the value is set to **1**, IP forwarding will be enabled.

Step 5: Verify the Automatic Switchover Between the Active and Standby ECSs

1. Restart the active and standby ECSs.
   a. Remotely log in to **ECS-HA1**.
      For details, see **How Do I Log In to My ECS?**
   b. Run the following command to restart **ECS-HA1**:
      reboot
   c. Repeat **1.a** to **1.b** to restart **ECS-HA2**.
2. Check whether the website on the active ECS can be accessed.
   a. Open a browser, enter the EIP address (**124.X.X.187**), and press **Enter**.
      If the following page is displayed, **ECS-HA1** is used as the active ECS and the website can be accessed.

      Figure 4-14 ECS-HA1 accessed

   b. Remotely log in to **ECS-HA1** and run the following command to check whether the virtual IP address is bound to the network interface (eth0) of **ECS-HA1**:
      ip addr show
      If information similar to the following is displayed, the virtual IP address (**192.168.0.177**) has been bound to the network interface (eth0) of **ECS-HA1**, and this ECS is the active one.

```
[root@ecs-ha1 ~]# ip addr show
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default
qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP group default
qlen 1000
    link/ether fa:16:3e:fe:56:19 brd ff:ff:ff:ff:ff:ff
    inet 192.168.0.195/24 brd 192.168.0.255 scope global noprefixroute dynamic eth0
       valid_lft 107898685sec preferred_lft 107898685sec
    inet 192.168.0.177/32 scope global eth0
       valid_lft forever preferred_lft forever
    inet6 fe80::f816:3eff:fefe:5619/64 scope link
       valid_lft forever preferred_lft forever
```

   c. Run the following command to disable Keepalived on **ECS-HA1**:
      systemctl stop keepalived.service
3. Check whether **ECS-HA2** becomes the active ECS.
   a. Remotely log in to **ECS-HA2** and run the following command to check whether the virtual IP address is bound to the network interface (eth0) of **ECS-HA2**:
      ip addr show

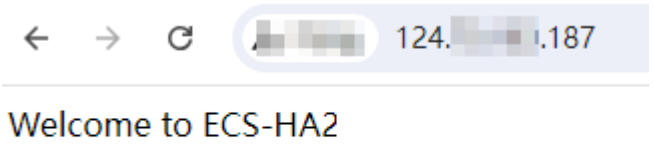
If information similar to the following is displayed, the virtual IP address (**192.168.0.177**) has been bound to the network interface (eth0) of **ECS-HA2**, and this ECS becomes the active one.

```
[root@ecs-ha2 ~]# ip addr show
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default
qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP group default
qlen 1000
    link/ether fa:16:3e:fe:56:3f brd ff:ff:ff:ff:ff:ff
    inet 192.168.0.233/24 brd 192.168.0.255 scope global noprefixroute dynamic eth0
       valid_lft 107898091sec preferred_lft 107898091sec
    inet 192.168.0.177/32 scope global eth0
       valid_lft forever preferred_lft forever
    inet6 fe80::f816:3eff:fefe:563f/64 scope link
       valid_lft forever preferred_lft forever
```

b. Open a browser, enter the EIP address (**124.X.X.187**), and press **Enter** to check whether the website on the active ECS (**ECS-HA2**) can be accessed.

   If the following page is displayed, **ECS-HA2** is used as the active ECS and the website can be accessed.

Figure 4-15 ECS-HA2 accessed



5 Elastic Network Interface and Supplementary Network Interface

5.1 Elastic Network Interface

5.1.1 Elastic Network Interface Overview

An elastic network interface (referred to as a network interface in this documentation) is a virtual network card. You can create and configure network interfaces and attach them to your instances (ECSs and BMSs) to obtain flexible and highly available network configurations.

Network Interface Types

- A primary network interface is created together with an instance by default, and cannot be detached from the instance.
- An extended network interface can be created on the **Network Interfaces** tab, and can be attached to or detached from an instance.

Application Scenarios

- Flexible migration

  You can detach a network interface from an instance and then attach it to another instance. The network interface retains its private IP address, EIP, and security group rules. In this way, service traffic on the faulty instance can be quickly migrated to the standby instance, implementing quick service recovery.

- Traffic management

  You can attach multiple network interfaces that belong to different subnets in a VPC to the same instance, and configure the network interfaces to carry the private network traffic, public network traffic, and management network traffic of the instance. You can configure access control policies and routing policies for each subnet, and configure security group rules for each network interface to isolate networks and service traffic.

Notes and Constraints

- The number of extended network interfaces that can be attached to an ECS is determined by the ECS specifications. For details, see **ECS Specifications**.
- Extended network interfaces cannot be used to directly access Huawei Cloud services, such as DNS. You can use VPCEP to access these services. For details, see **Buying a VPC Endpoint**.

5.1.2 Creating a Network Interface

Scenarios

A primary network interface is created together with an instance by default. You can perform the following operations to create extended network interfaces on the **Network Interfaces** console.

Notes and Constraints

An extended network interface created on the console can only be attached to an instance from the same VPC, but they can be associated with different security groups.

NOTE

If you create an extended network interface using an API, the interface can be attached to an instance from a different VPC.

Procedure

1. Go to the **network interface list page**.
2. Click **Create Network Interface**.
3. Configure parameters for the network interface, as shown in **Table 5-1**.

Table 5-1 Parameter descriptions

Parameter	Parameter Description	Example Value
Region	Region where the network interface is created. Select the region nearest to you to ensure the lowest latency possible.	CN-Hong Kong
Name	Name of the network interface. The name: • Can contain 1 to 64 characters. • Can contain letters, digits, underscores (_), hyphens (-), and periods (.).	networkInterface-891e
VPC	VPC where the network interface is created.	vpc-001

Parameter	Parameter Description	Example Value
Subnet	Subnet where the network interface is created.	subnet-001
IPv4 Address	How a private IPv4 address will be assigned to the network interface. There are two options: • **Automatically assign IP address**: The system assigns a private IPv4 from the subnet. • **Manually specify IP address**: You can specify an IP address. If you select **Manually specify IP address**, enter a private IPv4 address.	192.168.0.15
Security Group	Select the security group that will be associated with the network interface.	sg-001

4. Click **OK**.

5.1.3 Viewing the Basic Information About a Network Interface

Scenarios

You can view basic information about your network interface on the management console, including the name, ID, type, VPC, attached instance, and associated security groups.

Procedure

1. Log in to the management console.
2. Click in the upper left corner and select the desired region and project.
3. Click in the upper left corner and choose **Networking** > **Virtual Private Cloud**.

   The **Virtual Private Cloud** page is displayed.
4. In the navigation pane on the left, choose **Virtual Private Cloud** > **Network Interfaces**.
5. On the **Network Interfaces** page, click the private IP address of the target network interface.

Other Operations

On the network interface details page, you can also modify the following information:

- You can edit the network interface name, change IP addresses, and attach the network interface to or detach it from an instance.
- Enable or disable **Instance-dependent Deletion**.
  - **Instance-dependent Deletion** is disabled by default. The network interface will not be deleted if it is detached from the instance or if the instance is deleted. You can attach the network interface to another instance.
  - If **Instance-dependent Deletion** has been enabled, the network interface will be deleted after it is detached from the instance.

  The **Instance-dependent Deletion** option is only available in certain regions. See which regions support this option on the console.

5.1.4 Attaching a Network Interface to a Cloud Server

Scenarios

You can attach a network interface to an ECS or a BMS to achieve flexible and high-availability network configurations.

Procedure

1. Log in to the management console.
2. Click  in the upper left corner and select the desired region and project.
3. Click  in the upper left corner and choose **Networking** > **Virtual Private Cloud**.

   The **Virtual Private Cloud** page is displayed.
4. In the navigation pane on the left, choose **Virtual Private Cloud** > **Network Interfaces**.
5. In the network interface list, locate the row that contains the target network interface, click **Attach Instance** in the **Operation** column, and select the instance to be attached.
6. Click **OK**.

Related Operations

After a network interface is attached to an instance, it is recommended to enable NIC multi-queue to improve network performance. For details, see **Enabling NIC Multi-Queue**.

5.1.5 Binding an EIP to a Network Interface

Scenarios

You can bind an EIP to a network interface to achieve more flexible and scalable networks.

Each network interface has a private IP address. After the network interface is bound to an EIP, the network interface has both a private IP address and a public IP address. The binding between a network interface and an EIP will not change

even after the network interface is detached from an instance. After a network interface is migrated from one instance to another, its private IP address and EIP will be migrated together at the same time.

An instance can have multiple network interfaces attached. If each network interface has an EIP bound, the instance will have multiple EIPs and can provide flexible access services.

Procedure

1. Log in to the management console.
2. Click in the upper left corner and select the desired region and project.
3. Click in the upper left corner and choose **Networking** > **Virtual Private Cloud**.

   The **Virtual Private Cloud** page is displayed.
4. In the navigation pane on the left, choose **Virtual Private Cloud** > **Network Interfaces**.
5. In the network interface list, locate the row that contains the target network interface, click **Bind EIP** in the **Operation** column, and select the EIP to be bound.
6. Click **OK**.

5.1.6 Binding a Network Interface to a Virtual IP Address

Scenarios

You can bind a network interface to a virtual IP address so that you can access the instance attached to the network interface using the virtual IP address.

Only a network interface with an instance attached can be bound to a virtual IP address.

For more information about virtual IP addresses, see **Virtual IP Address Overview**.

Procedure

1. Log in to the management console.
2. Click in the upper left corner and select the desired region and project.
3. Click in the upper left corner and choose **Networking** > **Virtual Private Cloud**.

   The **Virtual Private Cloud** page is displayed.
4. In the navigation pane on the left, choose **Virtual Private Cloud** > **Network Interfaces**.
5. In the network interface list, locate the row that contains the target network interface, and choose **More** > **Bind Virtual IP Address** in the **Operation** column.

   The **IP Addresses** page will be displayed.

6. Locate the row that contains the target virtual IP address and click **Bind to Server** in the **Operation** column.
7. Select the server and network interface, and click **OK**.

5.1.7 Detaching a Network Interface from an Instance or Unbinding an EIP from a Network Interface

Scenarios

This section describes how to detach a network interface from an instance or unbind a network interface from an EIP.

Notes and Constraints

- If **Instance-dependent Deletion** is enabled for a network interface, the network interface will be deleted if it is detached from its instance.
  - Deleting a network interface will also delete any supplementary network interfaces and VLAN sub-interfaces attached to it.
  - Deleting a network interface will also unbind its EIP.
- If **Instance-dependent Deletion** is disabled for a network interface, the network interface will not be deleted if it is detached from its instance.

  If a network interface has an EIP bound, detaching the network interface from its instance will also unbind the EIP from the network interface.

NOTE

After an EIP is unbound from a network interface, if you do not release the EIP, the EIP will continue to be billed, but you can still bind it to other cloud resources.

Procedure

1. Log in to the management console.
2. Click in the upper left corner and select the desired region and project.
3. Click in the upper left corner and choose **Networking** > **Virtual Private Cloud**.

   The **Virtual Private Cloud** page is displayed.
4. In the navigation pane on the left, choose **Virtual Private Cloud** > **Network Interfaces**.
5. In the network interface list, locate the row that contains the target network interface, and click **Detach Instance** or **Unbind EIP** in the **Operation** column.
6. Click **OK**.

   If you no longer need an EIP, you can release the EIP after unbinding it.

5.1.8 Changing Security Groups That Are Associated with a Network Interface

Scenarios

You can change the security groups that are associated with a network interface on either the network interface list page or the network interface details page.

Procedure

Changing the security group associated with a network interface on the network interface list page

1. Log in to the management console.
2. Click  in the upper left corner and select the desired region and project.
3. Click  in the upper left corner and choose **Networking** > **Virtual Private Cloud**.

   The **Virtual Private Cloud** page is displayed.
4. In the navigation pane on the left, choose **Virtual Private Cloud** > **Network Interfaces**.
5. In the network interface list, locate the row that contains the target network interface, and choose **More** > **Change Security Group** in the **Operation** column.
6. On the **Change Security Group** page, select the security groups to be associated and click **OK**.

Changing the security group associated with a network interface on the network interface details page

1. Log in to the management console.
2. Click  in the upper left corner and select the desired region and project.
3. Click  in the upper left corner and choose **Networking** > **Virtual Private Cloud**.

   The **Virtual Private Cloud** page is displayed.
4. In the navigation pane on the left, choose **Virtual Private Cloud** > **Network Interfaces**.
5. Click the private IP address of the target network interface.
6. Click the **Associated Security Groups** tab. Then, click **Change Security Group**.
7. On the **Change Security Group** page, select the security groups to be associated and click **OK**.

Other Operations

On the **Associated Security Groups** tab, locate the target security group and click **Manage Rule** to manage security group rules. For details about how to configure security group rules, see **Adding a Security Group Rule**.

5.1.9 Deleting a Network Interface

Scenarios

This section describes how to delete a network interface.

Notes and Constraints

- A primary network interface is created together with an instance by default, and cannot be detached from the instance. If you want to delete a primary network interface, you need to delete its instance first, which will also delete the primary network interface.
- If you want to delete an extended network interface that is attached to an instance, **detach the interface from the instance** first.
- Deleting a network interface will also delete its supplementary network interfaces.
- Deleting a network interface will also unbind its EIP. You can choose to release the EIP if needed.

  If you do not release the EIP, the EIP will continue to be billed, but you can still bind it to other cloud resources.
- If a network interface has resources associated, deleting the interface will also delete any rules for associated resources that reference it.

  For example, if the next hop of a custom route in a VPC route table is a network interface, deleting the network interface will also delete the route.

Procedure

1. Log in to the management console.
2. Click  in the upper left corner and select the desired region and project.
3. Click  in the upper left corner and choose **Networking** > **Virtual Private Cloud**.

   The **Virtual Private Cloud** page is displayed.
4. In the navigation pane on the left, choose **Virtual Private Cloud** > **Network Interfaces**.
5. Locate the row that contains the network interface, click **More** in the **Operation** column, and click **Delete**.

   A confirmation dialog box is displayed.
6. Confirm the information and click **OK**.

5.2 Supplementary Network Interfaces

5.2.1 Supplementary Network Interface Overview

Supplementary network interfaces are a supplement to elastic network interfaces. If the number of elastic network interfaces that can be attached to your cloud server cannot meet your requirements, you can use supplementary network

interfaces, which can be attached to VLAN subinterfaces of elastic network interfaces.

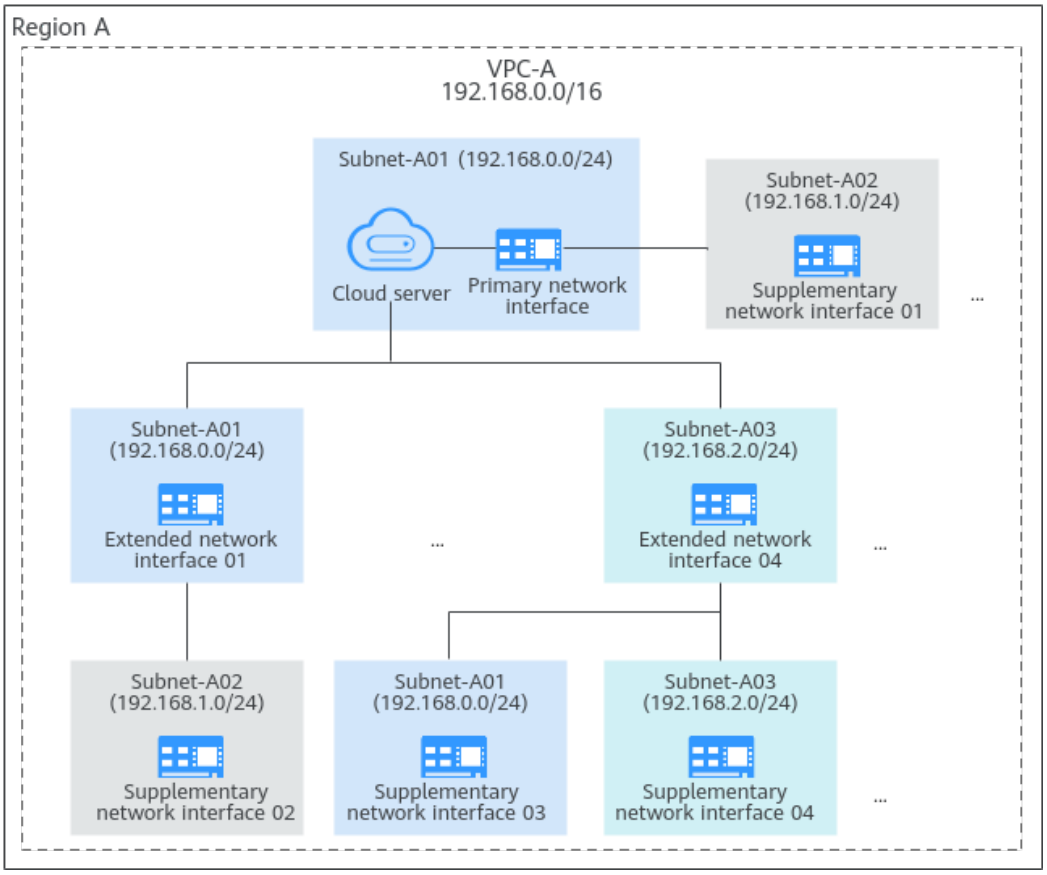
Application Scenarios

The number of elastic network interfaces that can be attached to each ECS is limited. If this limit cannot meet your requirements, you can attach supplementary network interfaces to elastic network interfaces.

- You can attach supplementary network interfaces that belong to different subnets in the same VPC to an ECS. Each supplementary network interface has its private IP address and EIP for private or Internet communication.
- You can security group rules for supplementary network interfaces for network isolation.

Supplementary network interfaces are attached to VLAN subinterfaces of elastic network interfaces, as shown in **Figure 5-1**. Supplementary network interfaces can be attached to both the primary network interface and extended network interfaces of a cloud server.

Figure 5-1 Supplementary network interface networking diagram



Constraints

- A maximum of 256 supplementary network interfaces can be attached to an ECS of certain flavors. The number of supplementary network interfaces that can be attached to an ECS varies by ECS flavor. ECS specifications that support supplementary network interfaces are as follows:

ECS: C7, S7, and M7 series. For details, see **ECS Specifications**.

Cloud container: c6ne

- An ECS cannot use Cloud-Init through the private IP addresses of its supplementary network interfaces.
- A supplementary network interface cannot have a virtual IP address bound.
- The flow logs of supplementary network interfaces cannot be collected separately. Their flow logs are generated together with their network interfaces.

5.2.2 Creating a Supplementary Network Interface

Scenarios

If the number of network interfaces attached to an instance exceeds the upper limit, you can attach supplementary network interfaces to the network interfaces, including the primary and extended network interfaces, of the instance. This helps you set up flexible and highly available networks.

Notes and Constraints

- Supplementary network interfaces must be in the same VPC as the network interface they are attached to, but they can be in different subnets and security groups.
- After supplementary network interfaces are created, you need to create VLAN subinterfaces on the network interface of the instance and configure corresponding rules by referring to **Configuring a Supplementary Network Interface**.

Creating a Supplementary Network Interface

1. Go to the **supplementary network interface list page**.
2. In the upper right corner of the page, click **Create Supplementary Network Interface**.
3. Configure the parameters based on **Table 5-2**.

Table 5-2 Parameter descriptions

Parameter	Description	Example Value
Region	Region where the supplementary network interface will be created. Select the region nearest to you to ensure the lowest latency possible.	CN-Hong Kong
Network Interface	Network interface that you want the supplementary network interface to attach to. Select an elastic network interface from the drop-down list.	--(172.16.0.145)

Parameter	Description	Example Value
VPC	VPC where the supplementary network interface will be created. The VPC of the network interface that the supplementary network interface is attached to is selected by default.	vpc-A
Subnet	Subnet where the supplementary network interface will be created. The supplementary network interface and its network interface can be in different subnets.	subnet-A01
Quantity	Number of supplementary network interfaces to be created.	1
Private IP Address	Whether to assign a private IPv4 address or IPv6 address to the supplementary network interface. There are two options: • **Private IPv4 network**: a private IPv4 address will be assigned. This option is selected by default and cannot be deselected. • **IPv6 network (Public and private network traffic)**: a private IPv6 address will be assigned. Both private and public IPv6 networks are supported. IPv6 is shown only when IPv6 is enabled for the subnet of the supplementary network interface.	IPv4
IPv4 Address	How a private IPv4 address will be assigned to the supplementary network interface. There are two options: • **Automatically assign IP address**: The system assigns an IP address from the subnet you have selected. • **Manually specify IP address**: You can specify an IP address. If you select **Manually specify IP address**, enter a private IPv4 address.	Automatically assign IP address

Parameter	Description	Example Value
IPv6 Address	How a IPv6 address will be assigned to the supplementary network interface if **IPv6 network (Public and private network traffic)** is selected for **Private IP Address**. There are two options: • **Automatically assign IP address**: The system assigns an IP address from the subnet. • **Manually specify IP address**: You can specify an IP address. If you select **Manually specify IP address**, enter a IPv6 address.	Automatically assign IP address
Security Group	Security group that the supplementary network interface will be associated with.	sg-001
Description (Optional)	Description of the supplementary network interface. The description can contain a maximum of 255 characters and cannot contain angle brackets (< or >).	-

4. Click **Create Now**.

NOTICE

To use a supplementary network interface, you need to create a VLAN subinterface by referring to **Configuring a Supplementary Network Interface**.

Configuring a Supplementary Network Interface

After a supplementary network interface is created, you need to create a VLAN subinterface for the network interface of the instance and configure a private IP address and default routes for the supplementary network interface.

Before doing so, you need to obtain:

- The information described in **Table 5-3** when you configure a supplementary network interface for a Linux ECS.
- The information described in **Table 5-3** and **Table 5-4** when you configure a supplementary network interface for a Windows ECS.

Table 5-3 Information about the supplementary network interface and subnet

Item	How to Obtain
VLAN ID MAC address Private IP address	1. In the supplementary network interface list, click the private IP address of the target supplementary network interface. The **Summary** page is displayed. 2. On the displayed page, check and record the following information: ● VLAN ID ● MAC address ● Private IP address
Subnet mask Gateway address	1. In the supplementary network interface list, locate the target supplementary network interface and click the subnet name in the **Network** column. The **Summary** page of the subnet is displayed. 2. On the displayed page, check and record the following information: ● Subnet mask: subnet mask of the IPv4 CIDR block. For example, if the IPv4 CIDR block is 192.168.0.0/24, the mask is 24. ● Subnet gateway: In the **Gateway and DNS Information** area, check the gateway address.

Table 5-4 Information about the network interface and subnet to which the supplementary network interface belongs

Item	How to Obtain
MAC address Private IP address	1. In the ECS list, click the name of the ECS attached to the network interface. The **Summary** page is displayed. 2. Switch to the **Network Interface** tab and click ⌄ to check and record the following information: ● MAC address ● Private IP address

Item	How to Obtain
Subnet mask Gateway address	1. In the network interface list, locate the target network interface and click the subnet name in the **Network** column. The **Summary** page of the subnet is displayed. 2. On the displayed page, check and record the following information: • Subnet mask: subnet mask of the IPv4 CIDR block. For example, if the IPv4 CIDR block is 192.168.0.0/24, the mask is 24. • Subnet gateway: In the **Gateway and DNS Information** area, check the gateway address.

Configuring a Supplementary Network Interface for a Linux ECS

The following describes how to create a VLAN subinterface on the network interface of a Linux ECS. CentOS 7.8 is used as an example. In this example, the information about the supplementary network interface and subnet is as follows:

- VLAN ID: 1937
- MAC address: fa:16:3e:6d:c5:5a
- Private IP address: 192.168.0.149
- Subnet mask: 24
- Subnet gateway address: 192.168.0.1

NOTE

This example describes how to configure the supplementary network interface for the primary network interface of an ECS. If you want to do the same thing for the extended network interface of the ECS, follow the similar steps.

1. Log in to the ECS.

   For details, see **Logging In to a Linux ECS**.

2. Run the following command to check and record the network interface name of the ECS:

   ifconfig

   Information similar to the following is displayed. In this example, the network interface name is **eth0**.

```
[root@ecs-subeni-linux ~]# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.0.125  netmask 255.255.255.0  broadcast 192.168.0.255
        inet6 fe80::f816:3eff:fe6d:c542  prefixlen 64  scopeid 0x20<link>
        ether fa:16:3e:6d:c5:42  txqueuelen 1000  (Ethernet)
        RX packets 78131  bytes 111604802 (106.4 MiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 8686  bytes 1422159 (1.3 MiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
...
```

3. Run the following command to create a VLAN subinterface on the network interface:

ip link add link *<network-interface-name>* **name** *<VLAN-subinterface-name>* **type vlan id** *<VLAN-ID-of-the-supplementary-network-interface>*

Variables in the preceding command are as follows:

- *network-interface-name*: the network interface name queried in 2. In this example, the name is **eth0**.
- *VLAN-subinterface-name*: Name the subinterface in the format of *<network-interface-name>***.***<VLAN-ID-of-the-supplementary-network-interface>*. In this example, the VLAN subinterface name is **eth0.1937**.
- *VLAN-ID-of-the-supplementary-network-interface*: In this example, the ID is **1937**.

Example command:

ip link add link eth0 name eth0.1937 type vlan id 1937

4. Run the following command to create a namespace:

   ip netns add *<namespace-name>*

   namespace-name: Name it in the format of **ns***<supplementary-network-interface-VLAN-ID>*. In this example, the name is **ns1937**.

   Example command:

   ip netns add ns1937

5. Run the following command to add the VLAN sub-interface to the namespace:

   ip link set *<VLAN-subinterface-name>* **netns** *<namespace-name>*

   Example command:

   ip link set eth0.1937 netns ns1937

6. Run the following command to change the MAC address of the VLAN subinterface to that of the supplementary network interface:

   ip netns exec *<namespace-name>* **ifconfig** *<VLAN-subinterface-name>* **hw ether** *<MAC-address-of-the-supplementary-network-interface>*

   Example command:

   ip netns exec ns1937 ifconfig eth0.1937 hw ether fa:16:3e:6d:c5:5a

7. Run the following command to enable the VLAN subinterface:

   ip netns exec *<namespace-name>* **ifconfig** *<VLAN-subinterface-name>* **up**

   Example command:

   ip netns exec ns1937 ifconfig eth0.1937 up

8. Run the following command to configure a private IP address for the VLAN subinterface:

   ip netns exec *<namespace-name>* **ip addr add** *<private-IP-address>* **dev** *<VLAN-subinterface-name>*

   private-IP-address: private IP address of the supplementary network interface/subnet mask. In this example, the value is **192.168.0.149/24**.

   Example command:

   ip netns exec ns1937 ip addr add 192.168.0.149/24 dev eth0.1937

9. Run the following command to configure the default route for the VLAN subinterface:

ip netns exec *<namespace-name>* **ip route add default via** *<gateway-address-of-the-subnet-where-the-supplementary-network-interface-is-created>*

Example command:

ip netns exec ns1937 ip route add default via 192.168.0.1

10. Check whether the supplementary network interface has worked.
    a. Run the following command to verify the connectivity between network interface **eth0** and the test ECS:

       ping *<private-IP-address-of-the-test-ECS>*

       Plan the same VPC and security group for the test ECS and the ECS with network interface **eth0** attached, so that the two iMetal servers can communicate with each other by default.

       Example command:

       ping 192.168.0.133

       If information similar to the following is displayed, the two ECSs can communicate with each other. If the communication is normal, proceed with **10.b**.

       ```
       [root@ecs-subeni-linux ~]# ping 192.168.0.133
       PING 192.168.0.133 (192.168.0.133) 56(84) bytes of data.
       64 bytes from 192.168.0.133: icmp_seq=1 ttl=64 time=0.302 ms
       64 bytes from 192.168.0.133: icmp_seq=2 ttl=64 time=0.262 ms
       ...
       --- 192.168.0.133 ping statistics ---
       2 packets transmitted, 2 received, 0% packet loss, time 999ms
       rtt min/avg/max/mdev = 0.262/0.282/0.302/0.020 ms
       ```

    b. Run the following command to verify the connectivity between the supplementary network interface and the test ECS:

       ip netns exec *<namespace-name>* **ping** *<private-IP-address-of-the-test-ECS>*

       Plan the same VPC and security group for the test ECS and the ECS with the supplementary network interface attached. This allows the two ECSs to communicate with each other by default.

       Example command:

       ip netns exec ns1937 ping 192.168.0.133

       If information similar to the following is displayed, the two ECSs can communicate with each other. This means the supplementary network interface has worked.

       ```
       [root@ecs-subeni-linux ~]# ip netns exec ns1937 ping 192.168.0.133
       PING 192.168.0.133 (192.168.0.133) 56(84) bytes of data.
       64 bytes from 192.168.0.133: icmp_seq=1 ttl=64 time=0.420 ms
       64 bytes from 192.168.0.133: icmp_seq=2 ttl=64 time=0.233 ms
       ...
       --- 192.168.0.133 ping statistics ---
       2 packets transmitted, 2 received, 0% packet loss, time 999ms
       rtt min/avg/max/mdev = 0.233/0.326/0.420/0.095 ms
       ```


NOTICE

- The route configured above is a temporary route that is applied once configured, and will be lost after ECS restarts. To avoid network disruptions, take step 11 to configure permanent routes instead.
- If the ECS needs to access a public domain name through the supplementary network interface, you need to take step 11 to configure DNS for that supplementary network interface and then restart the ECS.

11. Configure a permanent route and DNS for the supplementary network interface. The configuration will work after the ECS is restarted.
    a. Configure a permanent route for the supplementary network interface.
        i. Run the following command to open the **/etc/rc.local** file:

           vi /etc/rc.local
        ii. Press **i** to enter the editing mode.
        iii. Add the following content to the end of the file.

           The parameters and values must be the same as those in steps 3 to 9.

```
ip link add link eth0 name eth0.1937 type vlan id 1937
ip netns add ns1937
ip link set eth0.1937 netns ns1937
ip netns exec ns1937 ifconfig eth0.1937 hw ether fa:16:3e:6d:c5:5a
ip netns exec ns1937 ifconfig eth0.1937 up
ip netns exec ns1937 ip addr add 192.168.0.149/24 dev eth0.1937
ip netns exec ns1937 ip route add default via 192.168.0.1
```

        iv. Press **Esc** to exit and enter **:wq!** to save the configuration.
        v. Run the following command to assign execute permissions to the **/etc/rc.local** file:

           chmod +x /etc/rc.local

           NOTE

           If your operating system is Red Hat or EulerOS, run the following command after you perform 11.a.v:

           chmod +x /etc/rc.d/rc.local
    b. (Optional) Configure DNS for the supplementary network interface if the ECS needs to access the public domain name through the supplementary network interface.

       If DNS resolution is not required, take step 11.c to restart the ECS.
        i. Run the following command to go to the **/etc/sysconfig/network-scripts/** directory that stores the network interface configuration file:

           cd /etc/sysconfig/network-scripts/
        ii. Run the following command to modify the network interface configuration file:

           vi ifcfg-*<network-interface-name>*

           Network interface name: the name queried in 2. In this example, the name is **vi ifcfg-eth0**.
        iii. Press **i** to enter the editing mode.

iv. Add the following content to the end of the file.

114.114.114.114 is the public recursive DNS address.

```
DNS1=114.114.114.114
```

v. Press **Esc** to exit and enter **:wq!** to save the configuration.

c. Run the following command to restart the ECS:

reboot

d. Check whether the permanent route has worked by referring to **10**.

e. (Optional) If DNS is configured, check whether the DNS configuration has worked.

i. Bind an EIP to the supplementary network interface by referring to **Binding or Unbinding an EIP to or from a Supplementary Network Interface**.

ii. Run the following command to check whether the supplementary network interface can access the public domain name:

ip netns exec *<namespace-name>* **ping** *<public-domain-name>*

Example command:

ip netns exec ns1937 ping support.huaweicloud.com

If information similar to the following is displayed, the DNS configuration has worked.

```
[root@ecs-subeni-linux ~]# ip netns exec ns1937 ping support.huaweicloud.com
PING support.huaweicloud.com (36.150.72.70) 56(84) bytes of data.
64 bytes from 36.150.72.70 (36.150.72.70): icmp_seq=1 ttl=54 time=2.68 ms
64 bytes from 36.150.72.70 (36.150.72.70): icmp_seq=2 ttl=54 time=2.61 ms
64 bytes from 36.150.72.70 (36.150.72.70): icmp_seq=3 ttl=54 time=2.60 ms
^C
--- support.huaweicloud.com ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 3015ms
rtt min/avg/max/mdev = 2.604/2.633/2.681/0.068 ms
```

12. (Optional) Remotely log in to the ECS using the private IP address of the supplementary network interface.

a. Add an inbound rule to allow traffic over SSH port 22 to the security group associated with the supplementary network interface.

For details, see **Adding a Security Group Rule**.

Table 5-5 A security group rule that allows traffic over SSH port 22

Direction	Priority	Action	Type	Protocol & Port	Source
Inbound	1	Allow	IPv4	TCP: 22	Set the IP address based on service requirements. For example, to remotely log in to the ECS from a local PC, set the source to the IP address of the local PC.

b. Run the following command to check whether port 22 in the namespace is listened on:

ip netns exec *<namespace-name>* **netstat -antp | grep 22**

Example command:

ip netns exec ns1937 netstat -antp | grep 22

- If the command output is empty, port 22 in the namespace is not listened on. Go to **12.c**.
- If information similar to the following is displayed, port 22 is listened on. No further action is required.

```
[root@ecs-subeni-linux ~]# ip netns exec ns1937 netstat -antp | grep 22
tcp        0      0 0.0.0.0:22          0.0.0.0:*           LISTEN      2797/sshd
tcp6       0      0 :::22               :::*                LISTEN      2979/sshd
```

c. Run the following command to start the SSH service and enable listening port 22:

ip netns exec *<namespace-name>* **/sbin/sshd**

Example command:

ip netns exec ns1937 /sbin/sshd

d. Run the following command to check whether port 22 in the namespace is listened on:

ip netns exec *<namespace-name>* **netstat -antp | grep 22**

Example command:

ip netns exec ns1937 netstat -antp | grep 22

If information similar to the following is displayed, port 22 is listened on:

```
[root@ecs-subeni-linux ~]# ip netns exec ns1937 netstat -antp | grep 22
tcp        0      0 0.0.0.0:22          0.0.0.0:*           LISTEN      2797/sshd
tcp6       0      0 :::22               :::*                LISTEN      2979/sshd
```

13. (Optional) Allow traffic over HTTP port 80 of the supplementary network interface if the ECS needs to provide the web service through the supplementary network interface.

a. Add an inbound rule to allow traffic over HTTP port 80 to the security group associated with the supplementary network interface.

For details, see **Adding a Security Group Rule**.

Table 5-6 A security group rule that allows traffic over HTTP port 80

Direction	Priority	Action	Type	Protocol & Port	Source
Inbound	1	Allow	IPv4	TCP: 80	0.0.0.0/0 Allows any IP address to access the supplementary network interface over port 80.

b. Run the following command to check whether port 80 in the namespace is listened on:

ip netns exec *<namespace-name>* **netstat -antp | grep 80**

Example command:

ip netns exec ns1937 netstat -antp | grep 80

- If the command output is empty, port 80 in the namespace is not listened on. In this case, enable port 80 for the web service.
- If information similar to the following is displayed, port 80 is listened on. No further operation is required.

```
[root@ecs-subeni-linux ~]# ip netns exec ns1937 netstat -antp | grep 80
tcp6       0      0 :::80                   :::*                    LISTEN      ...
```

Configuring a Supplementary Network Interface for a Windows ECS

The following describes how to create a VLAN subinterface on the network interface of a Windows ECS. Windows Server 2019 Standard 64bit is used as an example. In this example, the information about the supplementary network interface, primary network interface, and subnet is as follows:

- Supplementary network interface
  - VLAN ID: 2242
  - MAC address: fa:16:3e:6d:c5:db
  - Private IP address: 192.168.0.22
  - Subnet mask: 24 (255.255.255.0)
  - Subnet gateway address: 192.168.0.1
- Network interface
  - MAC address: fa:16:3e:6d:c5:d5
  - Private IP address: 192.168.0.16
  - Subnet mask: 24 (255.255.255.0)
  - Subnet gateway address: 192.168.0.1

  NOTE

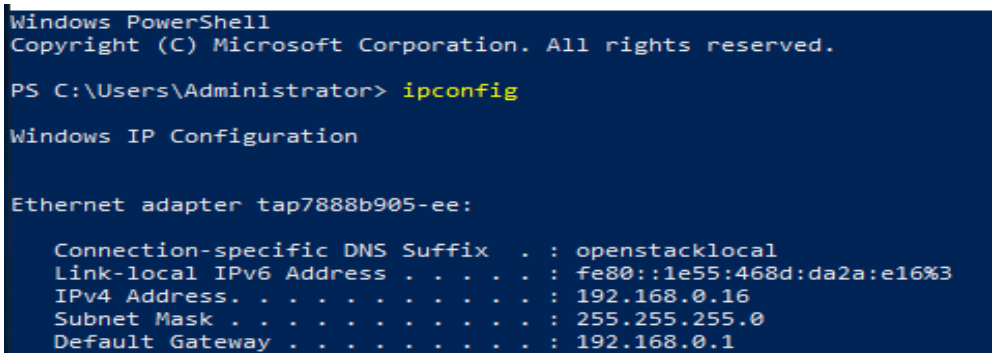
  This example describes how to configure the supplementary network interface for the primary network interface of an ECS. If you want to do the same thing for the extended network interface of the ECS, follow the similar steps.

1. Log in to the ECS.

   For details, see **Login Overview (Windows)**.
2. Enter **Windows PowerShell** in the search box in the lower left corner of the desktop and press **Enter**.
3. On the displayed window, run the following command to query the Ethernet adapter information of the network interface:

   ipconfig

   Information similar to the following is displayed. In this example, the Ethernet adapter name is **tap7888b905-ee**.



4. Create a bond group.

   a. Run the following command to create a bond group for the custom VLAN:

      New-NetLbfoTeam -Name *<bond-group-name>* **-TeamMembers "***<Ethernet-adapter-name-of-the-network-interface>***" -TeamingMode SwitchIndependent -LoadBalancingAlgorithm IPAddresses -Confirm:$false**

      Variables in the preceding command are as follows:

      - *bond-group-name*: the bond group name of the custom VLAN. In this example, the bond group name is **Team1**.
      - *Ethernet-adapter-name-of-the-network-interface*: information queried in **3**. In this example, the name is **tap7888b905-ee**.

      Example command:

      New-NetLbfoTeam -Name Team1 -TeamMembers "tap7888b905-ee" -TeamingMode SwitchIndependent -LoadBalancingAlgorithm IPAddresses -Confirm:$false

      Information similar to the following is displayed.

      ```
      PS C:\Users\Administrator> New-NetLbfoTeam -Name Team1 -TeamMembers "tap7888b905-ee" -TeamingMode SwitchIndependent -LoadBalancingAlgorithm IPAddresses -Confirm:$false

      Name                   : Team1
      Members                : tap7888b905-ee
      TeamNics               : Team1
      TeamingMode            : SwitchIndependent
      LoadBalancingAlgorithm : IPAddresses
      Status                 : Up
      ```

   b. Run the following commands to query the bond group you have created:

      Get-NetLbfoTeamMember

      Information similar to the following is displayed.

      ```
      PS C:\Users\Administrator> Get-NetLbfoTeamMember


      Name                   : tap7888b905-ee
      InterfaceDescription   : Red Hat VirtIO Ethernet Adapter
      Team                   : Team1
      AdministrativeMode     : Active
      OperationalStatus      : Active
      TransmitLinkSpeed(Gbps) : 100
      ReceiveLinkSpeed(Gbps)  : 100
      FailureReason          : NoFailure
      ```

      Get-NetAdapter

      Information similar to the following is displayed:

      ```
      PS C:\Users\Administrator> Get-NetAdapter

      Name                      InterfaceDescription                    ifIndex Status       MacAddress             LinkSpeed
      ----                      --------------------                    ------- ------       ----------             ---------
      Team1                     Microsoft Network Adapter Multiplexo...       9 Up           FA-16-3E-6D-C5-D5      100 Gbps
      tap7888b905-ee            Red Hat VirtIO Ethernet Adapter               3 Up           FA-16-3E-6D-C5-D5      100 Gbps
      ```

5. Configure a custom VLAN network.

   a. Run the following command to create a VLAN subinterface:

      Add-NetLbfoTeamNIC -Team "*<bond-group-name>***" -VlanID** *<VLAN-ID-of-the-supplementary-network-interface>* **-Confirm:$false**

Example command:

Add-NetLbfoTeamNIC -Team "Team1" -VlanID 2242 -Confirm:$false

Information similar to the following is displayed:

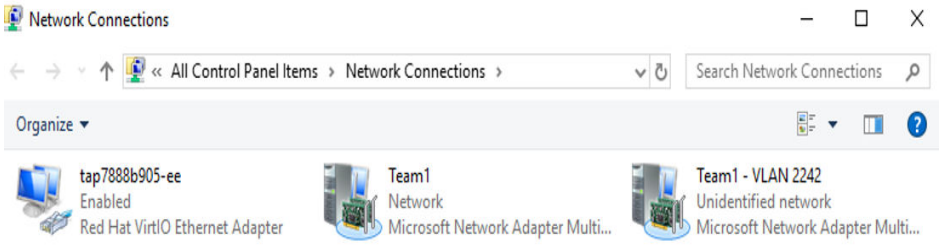
```
PS C:\Users\Administrator> Add-NetLbfoTeamNIC -Team "Team1" -VlanID 2242 -Confirm:$false


Name                      : Team1 - VLAN 2242
InterfaceDescription      : Microsoft Network Adapter Multiplexor Driver #2
Team                      : Team1
VlanID                    : 2242
Primary                   : False
Default                   : False
TransmitLinkSpeed(Gbps)   : 100
ReceiveLinkSpeed(Gbps)    : 100
```

b. Run the following command to open the **Network Connections** page:

ncpa.cpl

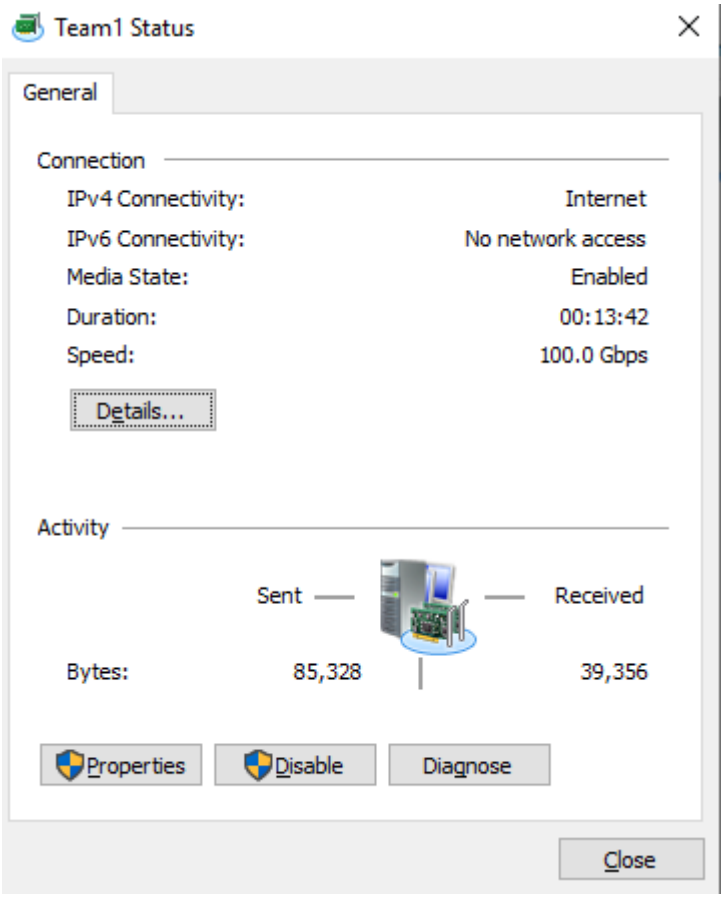
On the displayed page, **Team1** is the bond group created in **4.a**, and **Team1 – VLAN 2242** is the VLAN subinterface created in **5.a**.



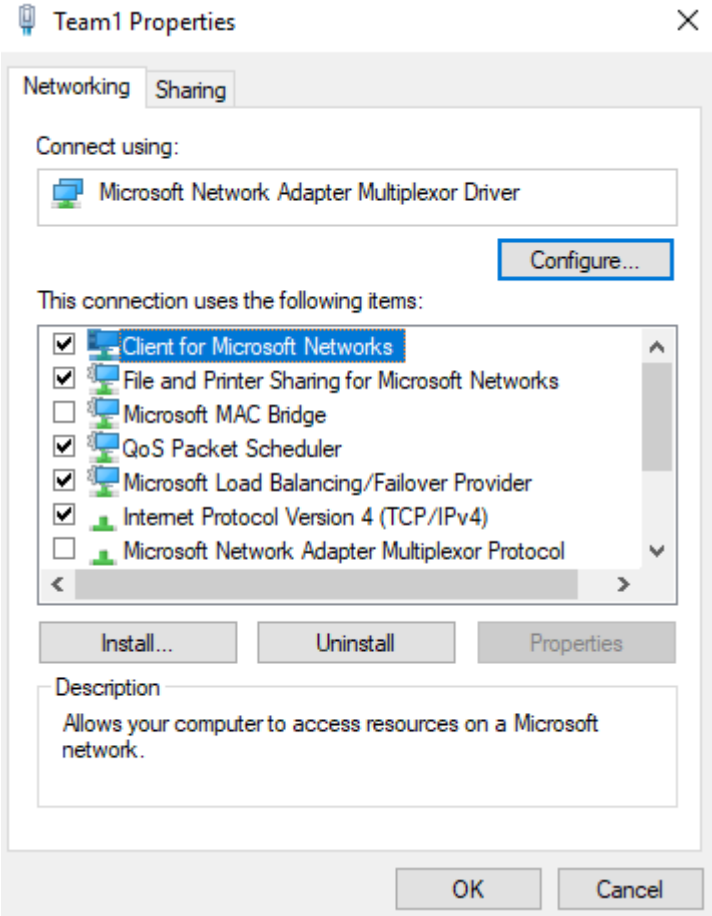
6. Configure the network for the network interface.

a. On the **Network Connections** page, double-click **Team1**.

The **Team1 Status** page is displayed.

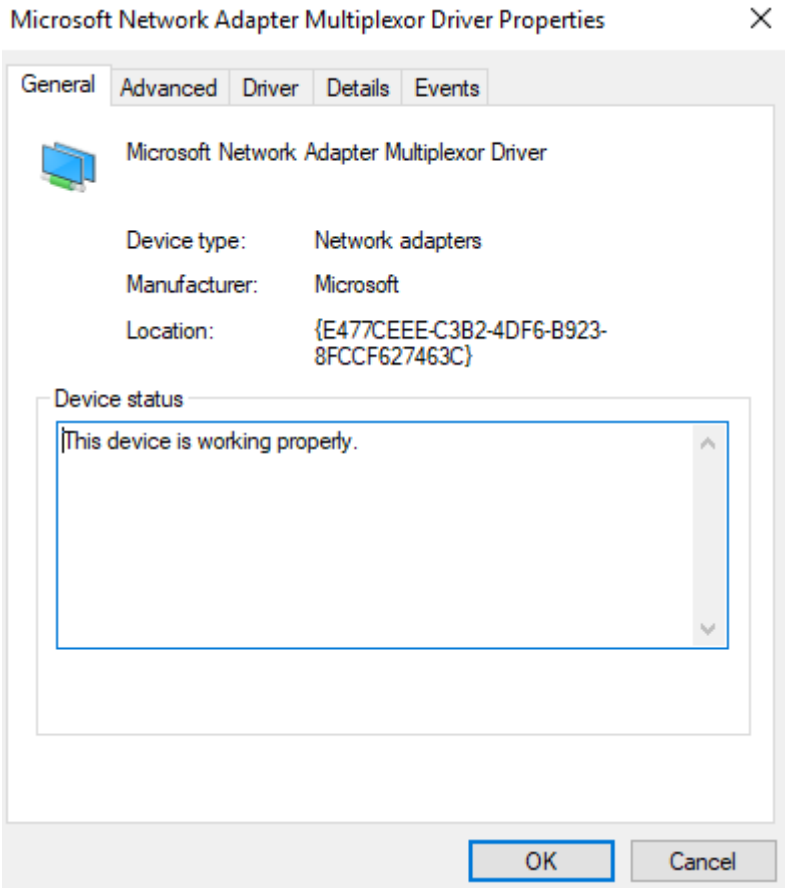


b. On the **Team1 Status** page, click **Properties**.
The **Team1 Properties** page is displayed.



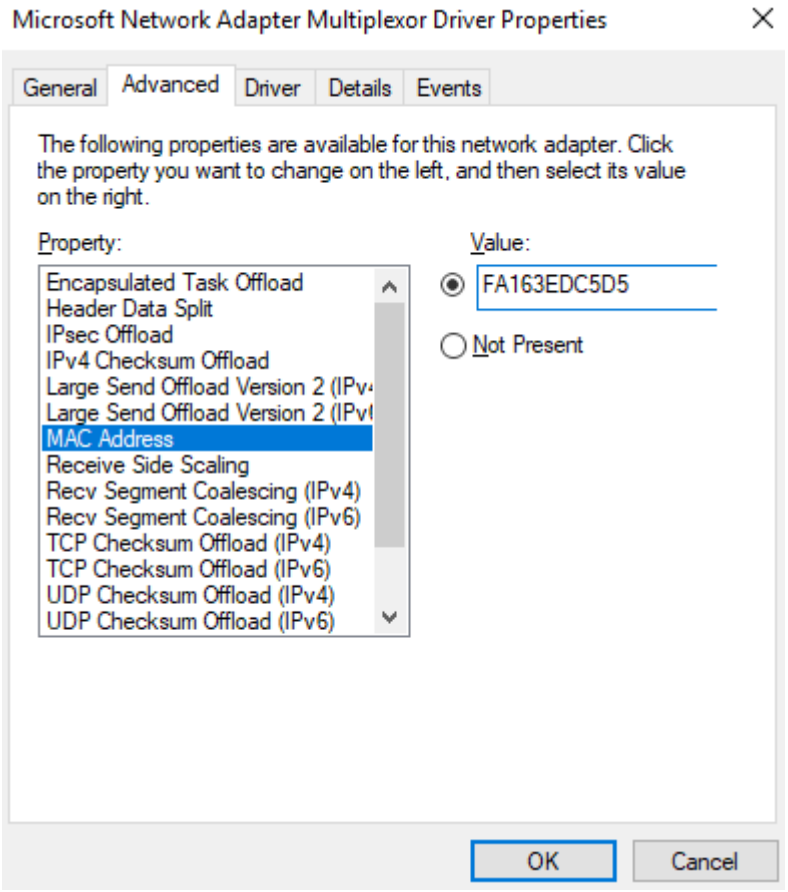
c. On the **Team1 Properties** page, click **Configure...**.

   The **Microsoft Network Adapter Multiplexor Driver Properties** page is displayed.



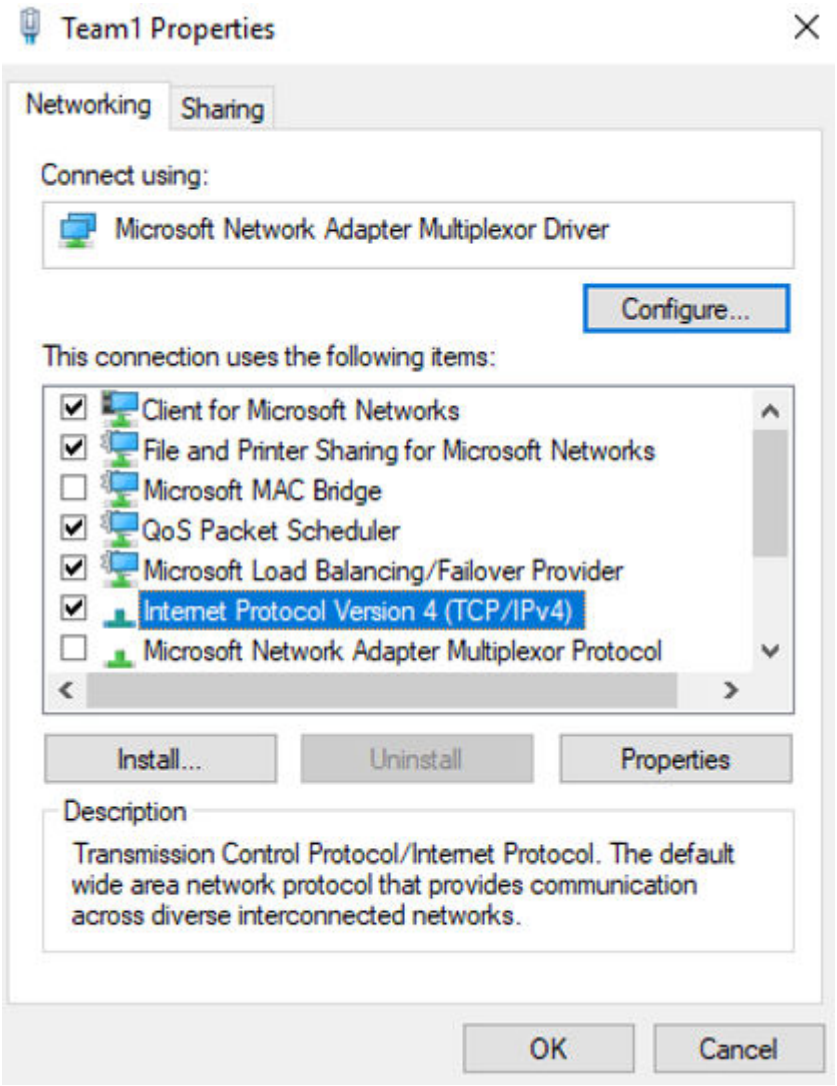
d. On the **Microsoft Network Adapter Multiplexor Driver Properties** page, choose the **Advanced** tab, click **MAC Address**, enter the MAC address of the network interface, and click **OK**.

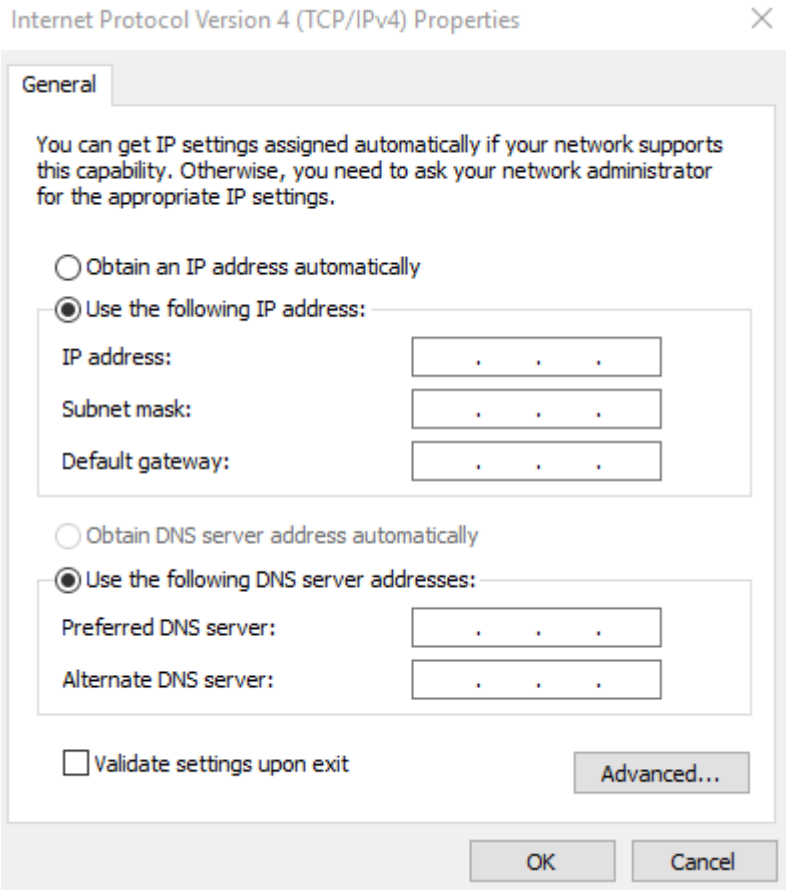
   When entering the MAC address, remove the colons (:) and use the uppercase letters. For example, if the MAC address of the network interface is **fa:16:3e:6d:c5:d5**, enter **FA163E6DC5D5**.



e. Return to the **Team1 Properties** page, double-click **Internet Protocol Version 4 (TCP/IPv4)**.

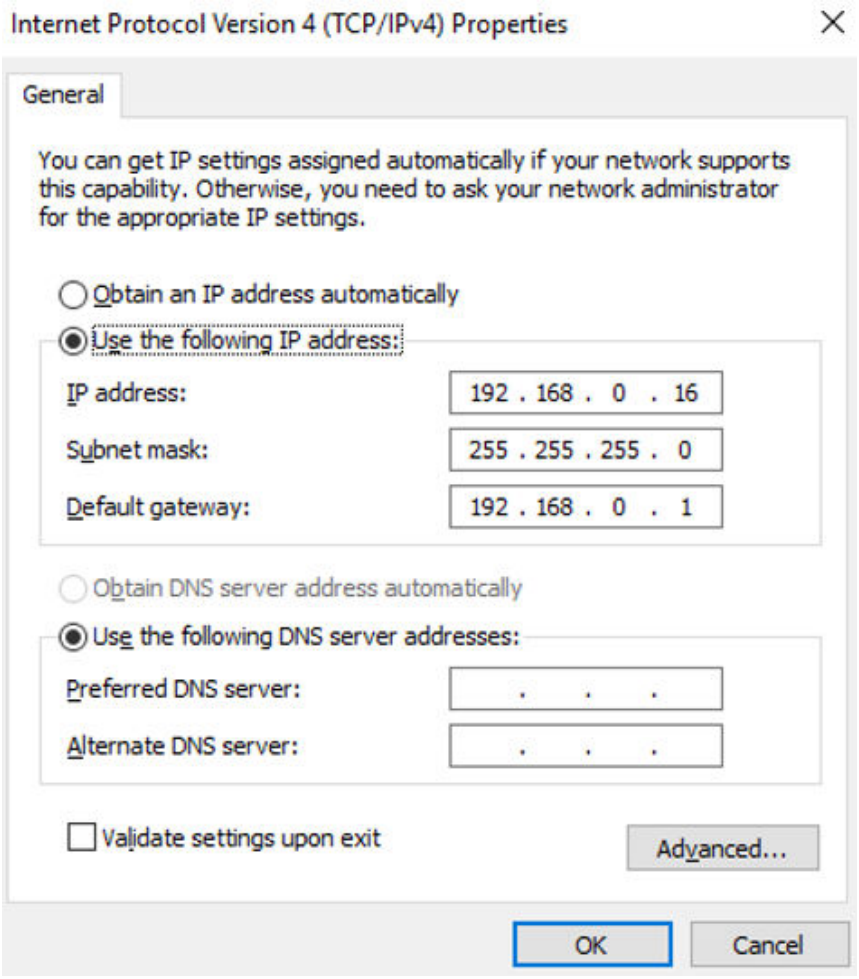
   The **Internet Protocol Version 4 (TCP/IPv4) Properties** page is displayed.



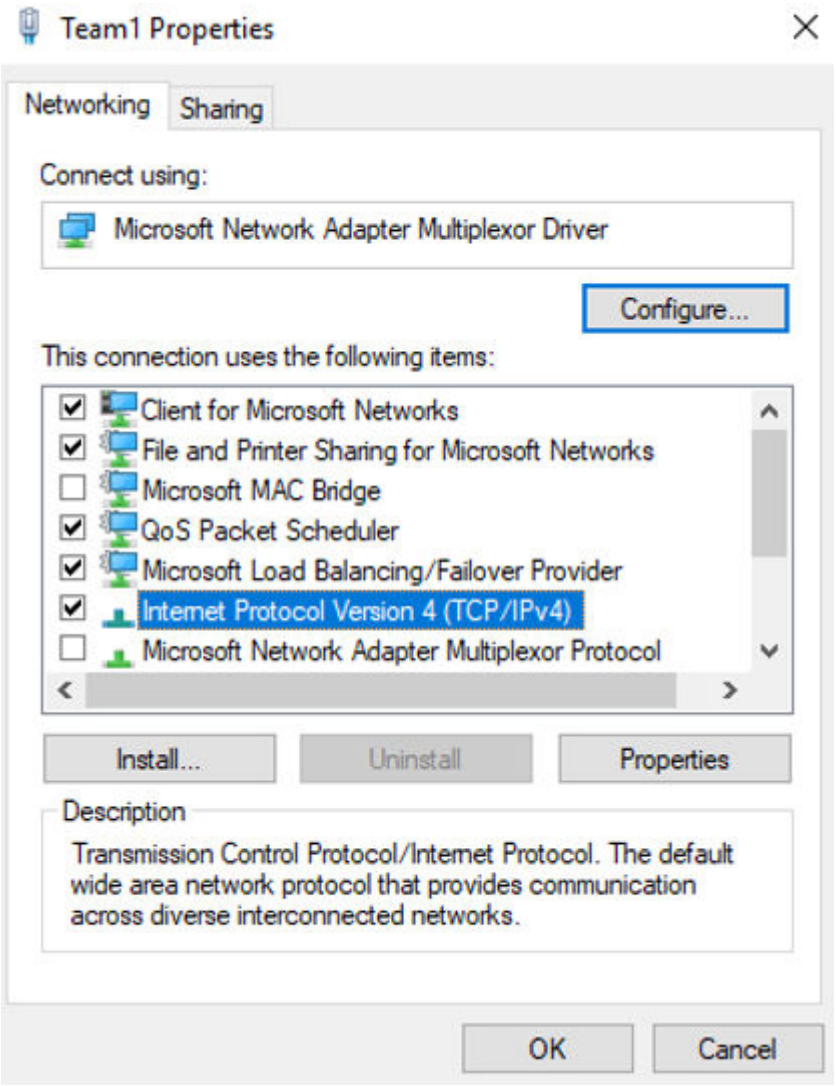


f. On the **Internet Protocol Version 4 (TCP/IPv4) Properties** page, configure the network information of the network interface and click **OK**.

- Select **Use the following IP address:**.
- **IP address**: Enter the private IP address of the network interface. In this example, the private IP address is **192.168.0.16**.
- **Subnet mask**: Enter the mask of the subnet where the network interface is created. In this example, the mask is **255.255.255.0**.
- **Default gateway**: Enter the gateway of the subnet where the network interface is created. In this example, the gateway is **192.168.0.1**.



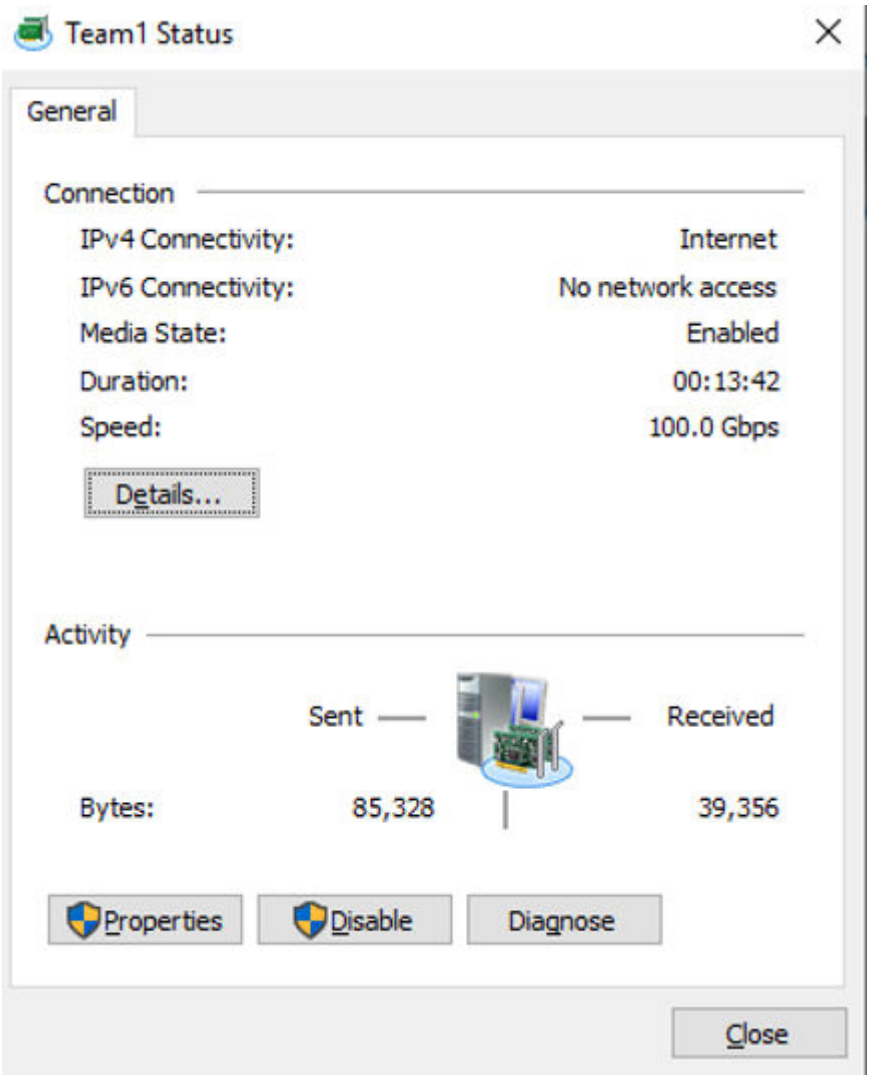
g. On the **Team1 Properties** page, click **OK** to save the settings.

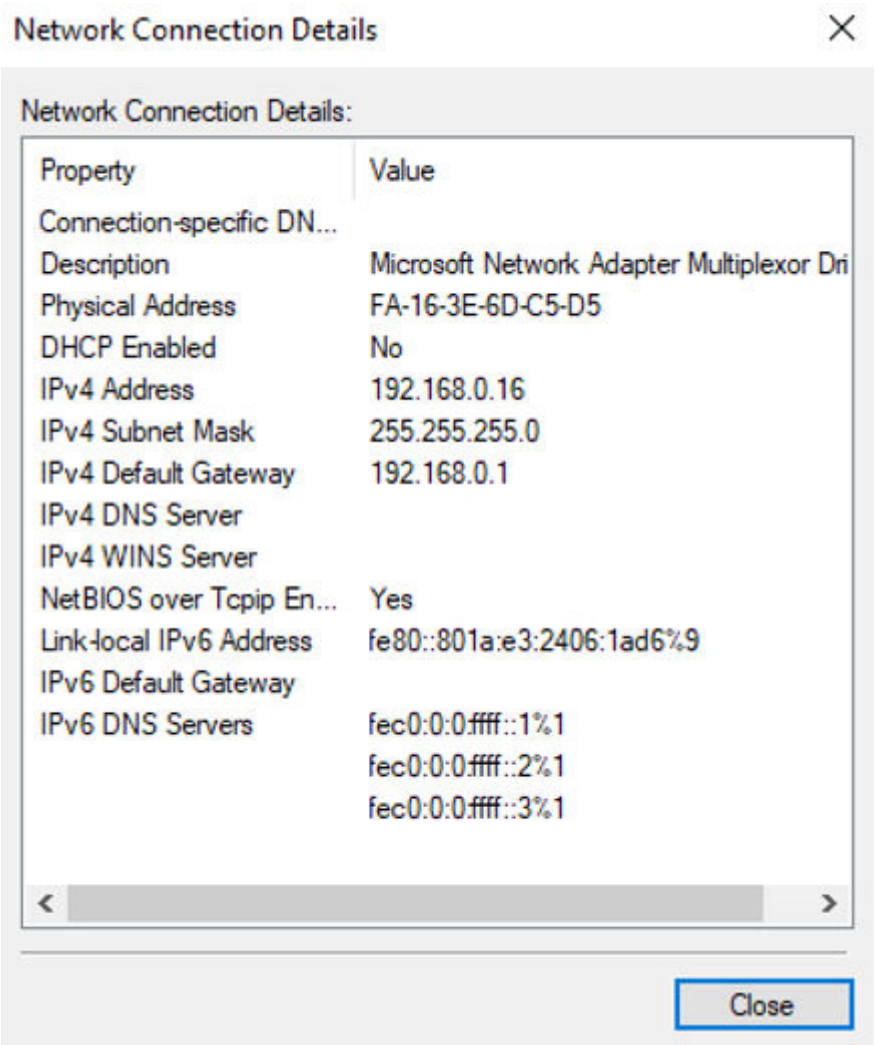


h. Return to the **Team1 Status** page and click **Details...**.

On the **Network Connection Details** page, check whether the following information is correctly configured:

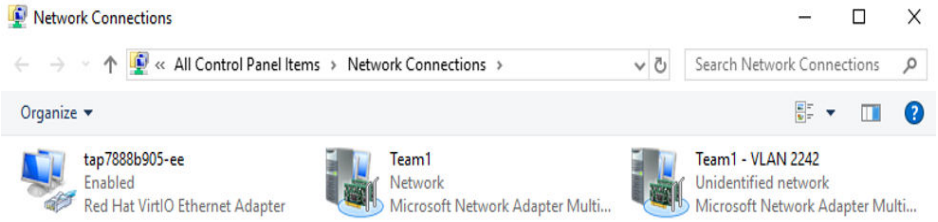
- **Physical Address**: MAC address of the network interface.
- **IPv4 Address**: the private IP address of the network interface.
- **IPv4 Subnet Mask**: the mask of the subnet where the network interface is created.
- **IPv4 Default Gateway**: the gateway of the subnet where the network interface is created.





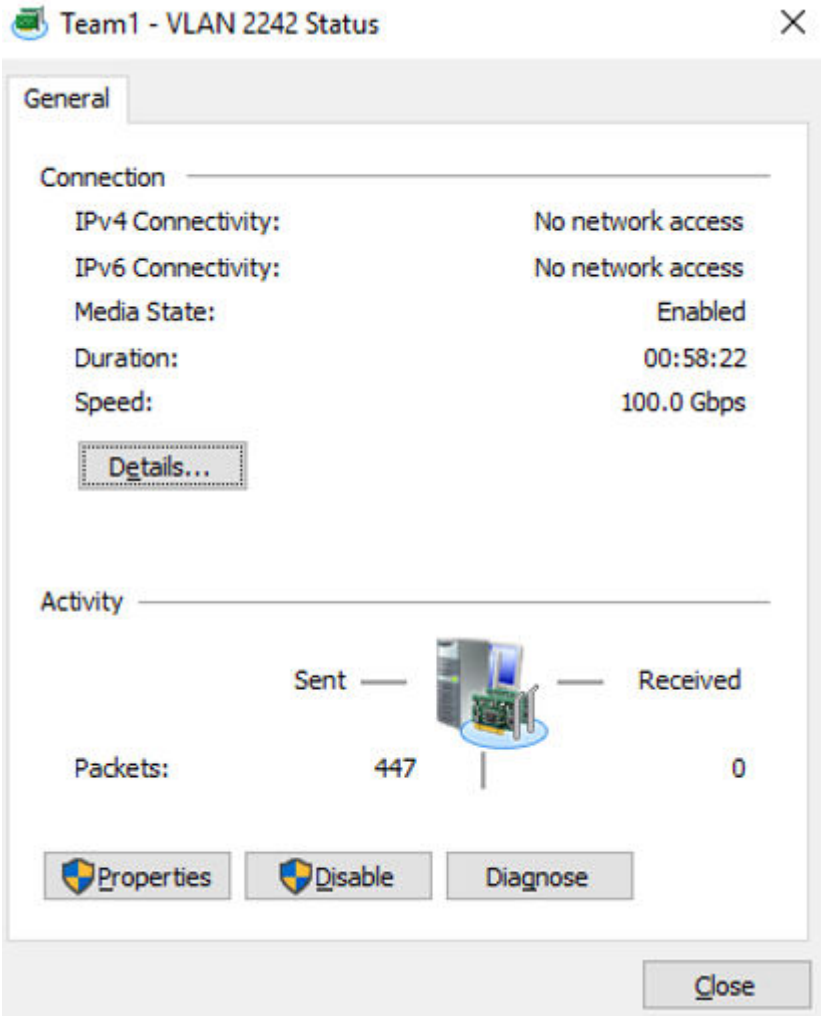
i. Check the settings and click **Close**.

The **Network Connections** page is displayed.

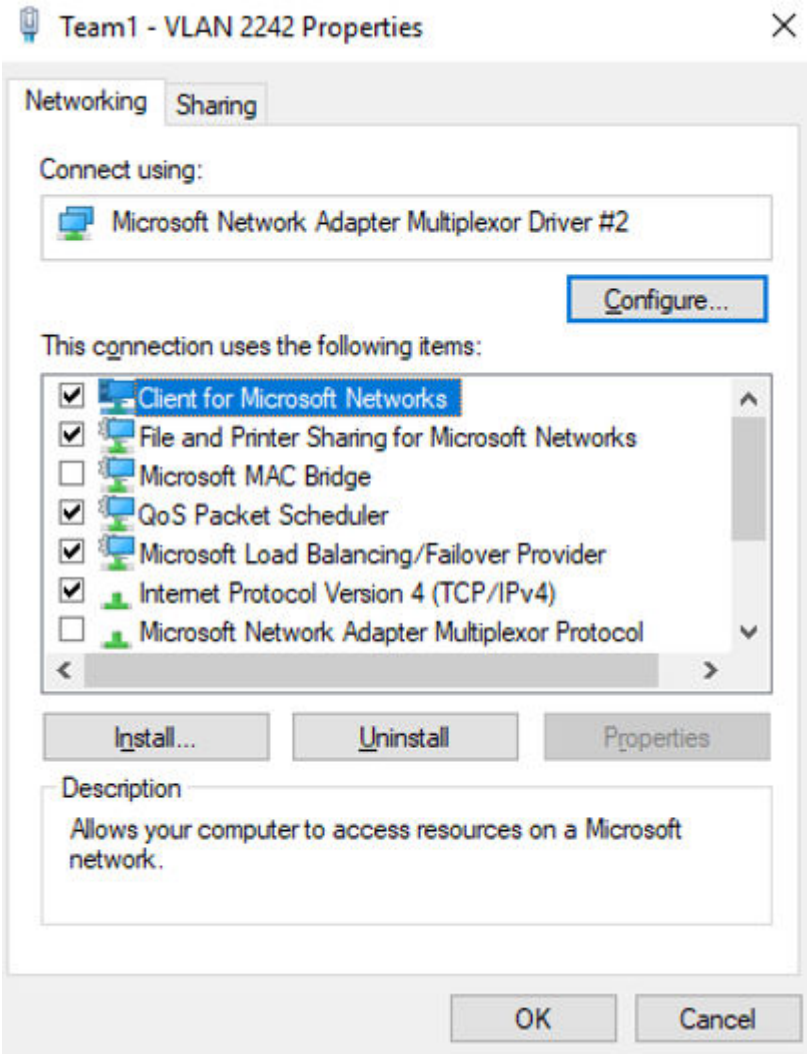


7. Configure the network for the supplementary network interface.
   a. On the **Network Connections** page, double-click **Team1 - VLAN 2242**.

      The **Team1 - VLAN 2242 Status** page is displayed.

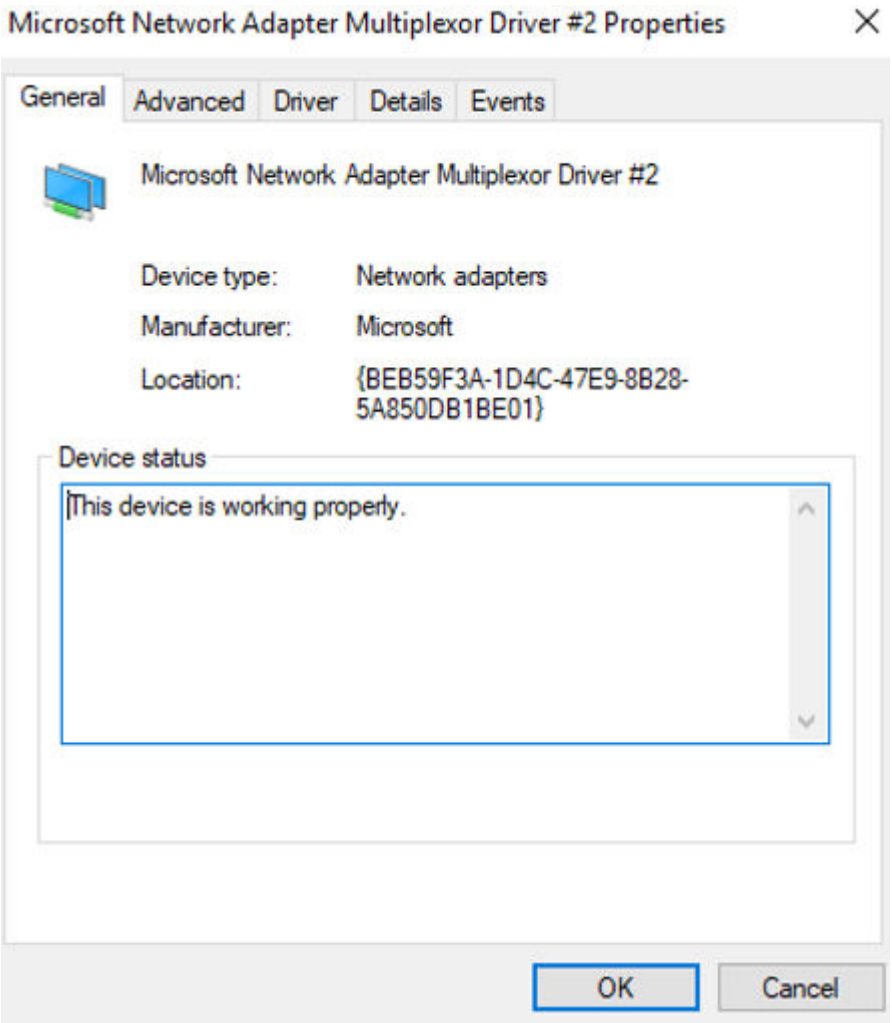


b. On the **Team1 - VLAN 2242 Status** page, click **Properties**.
The **Team1 - VLAN 2242 Properties** page is displayed.



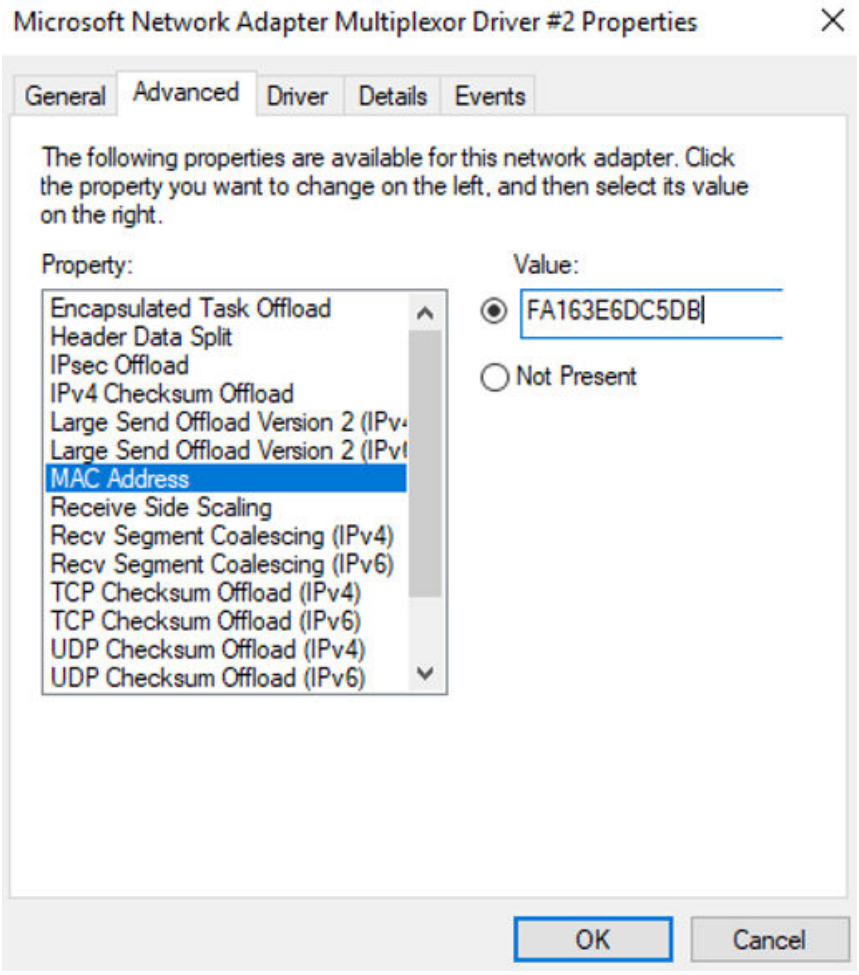
c. On the **Team1 - VLAN 2242 Properties** page, click **Configure...**.

The **Microsoft Network Adapter Multiplexor Driver #2 Properties** page is displayed.



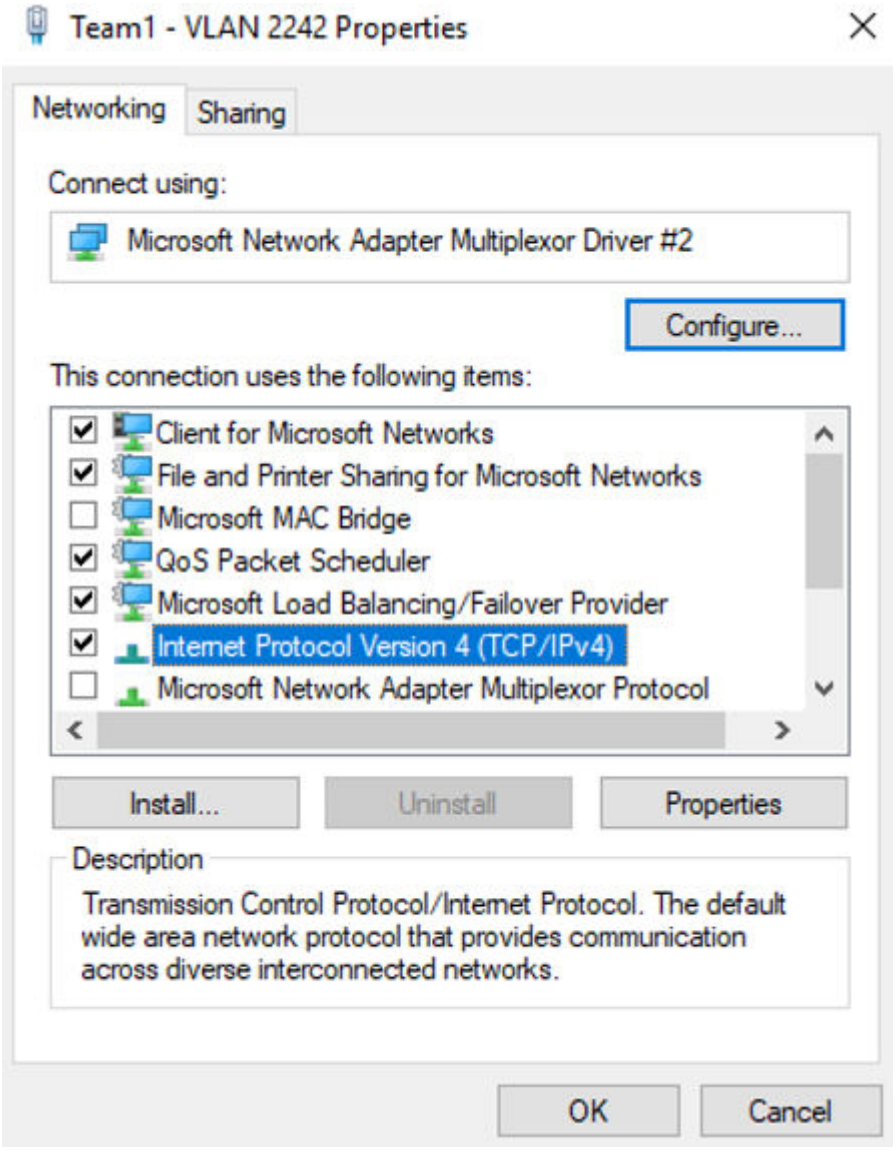
d. On the **Microsoft Network Adapter Multiplexor Driver #2 Properties** page, choose the **Advanced** tab, click **MAC Address**, enter the MAC address of the supplementary network interface, and click **OK**.

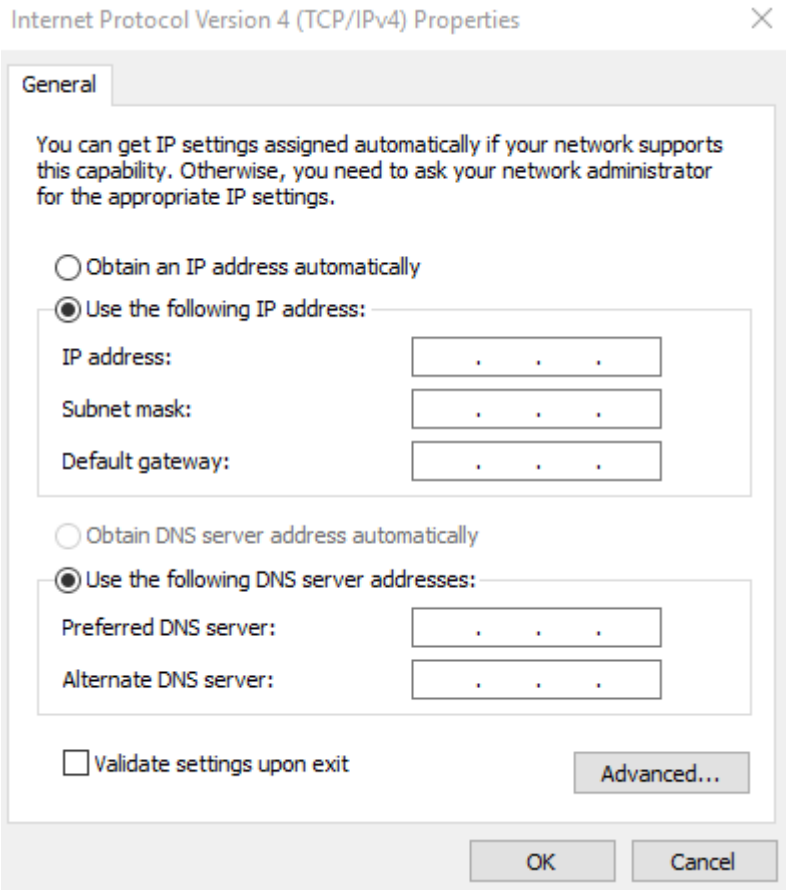
   When entering the MAC address, remove the colons (:) and use the uppercase letters. For example, if the MAC address of the supplementary network interface is **fa:16:3e:6d:c5:db**, enter **FA163E6DC5DB**.



e. Return to the **Team1 - VLAN 2242 Properties** page, double-click **Internet Protocol Version 4 (TCP/IPv4)**.

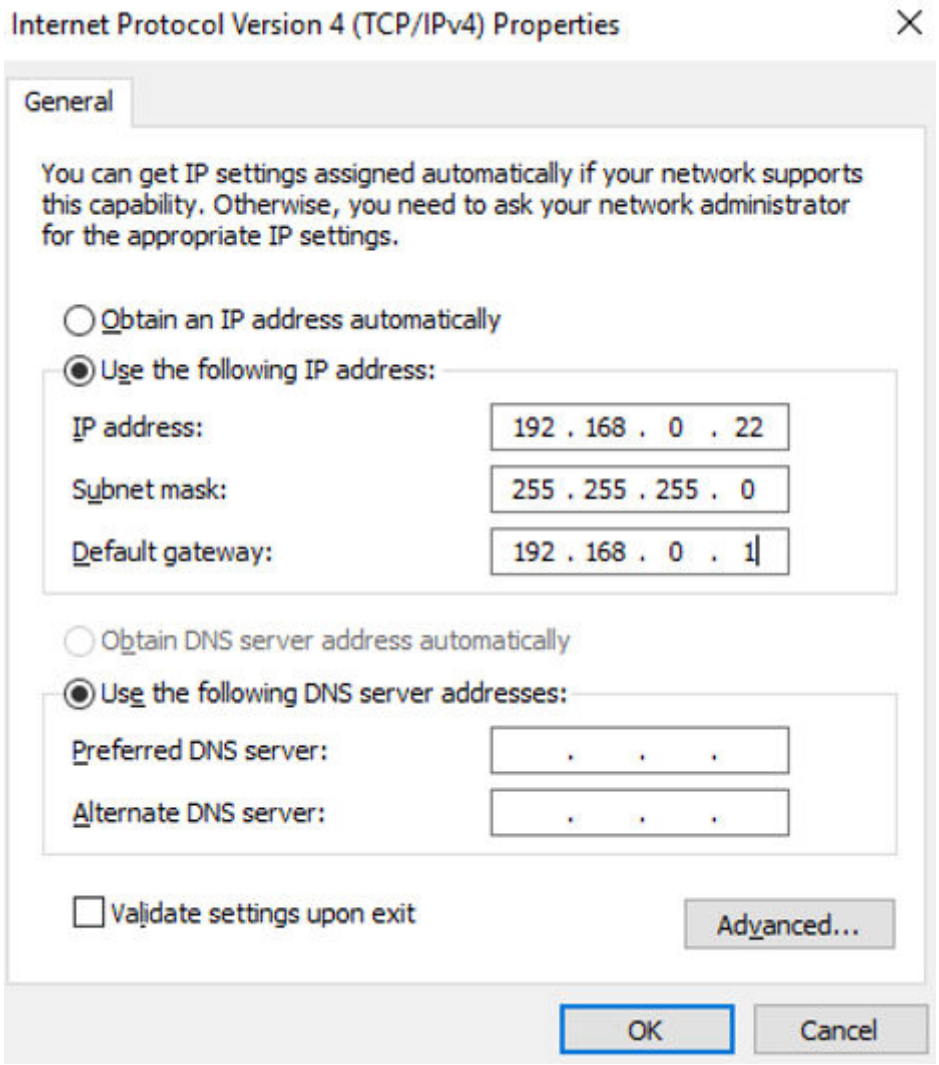
   The **Internet Protocol Version 4 (TCP/IPv4) Properties** page is displayed.



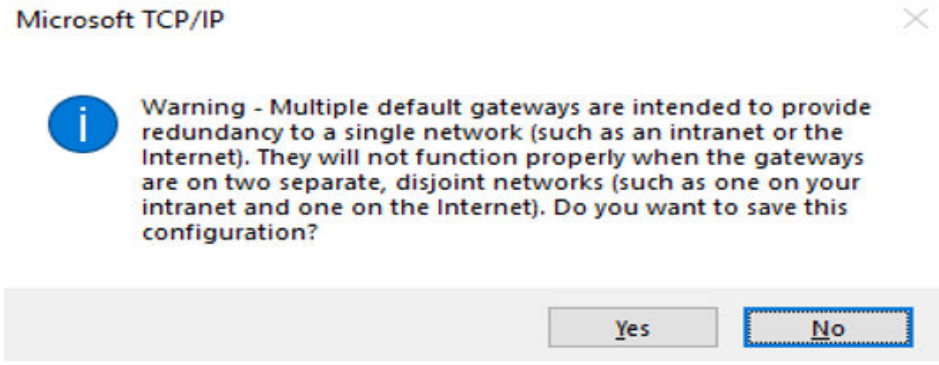


f. On the **Internet Protocol Version 4 (TCP/IPv4) Properties** page, configure the network information of the supplementary network interface and click **OK**.

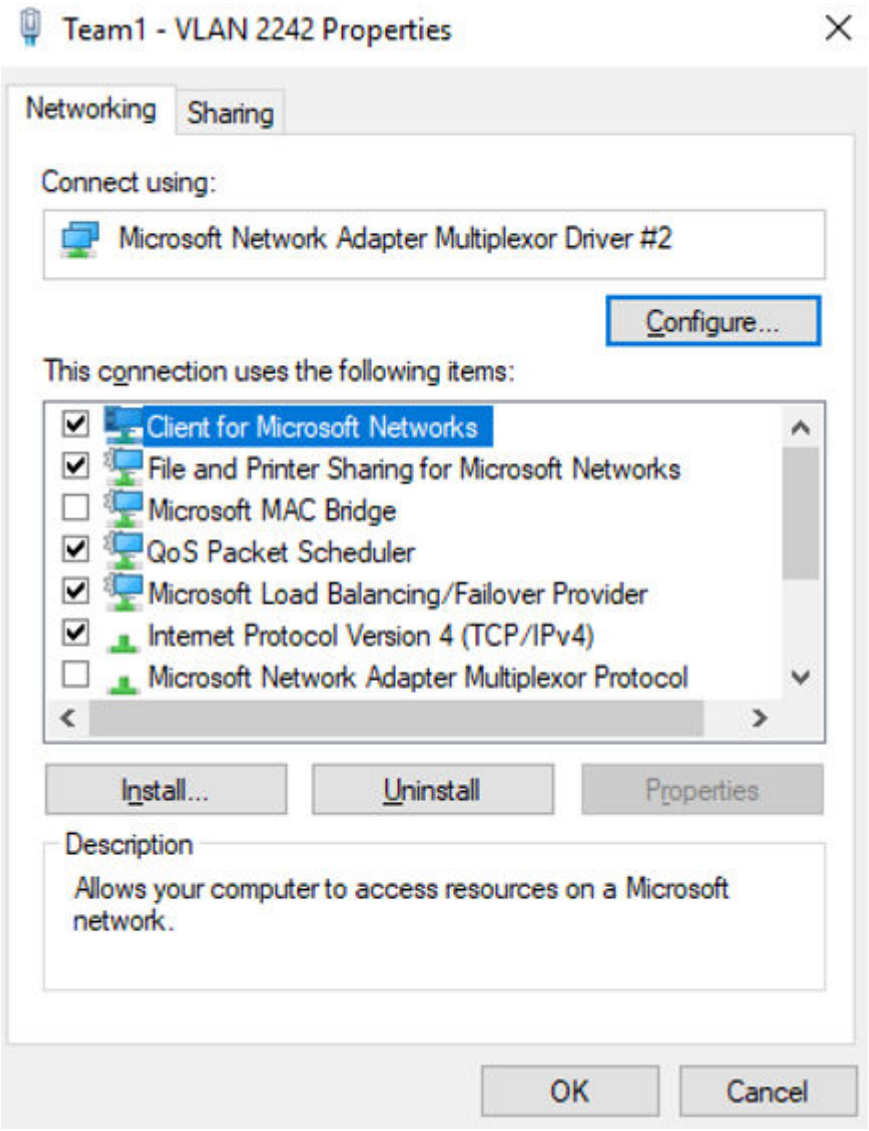
- Select **Use the following IP address:**.
- **IP address**: Enter the private IP address of the supplementary network interface. In this example, the private IP address is **192.168.0.22**.
- **Subnet mask**: Enter the mask of the subnet where the supplementary network interface is created. In this example, the mask is **255.255.255.0**.
- **Default gateway**: Enter the gateway of the subnet where the supplementary network interface is created. In this example, the gateway is **192.168.0.1**.



If the following warning is displayed, click **Yes** to close the dialog box.



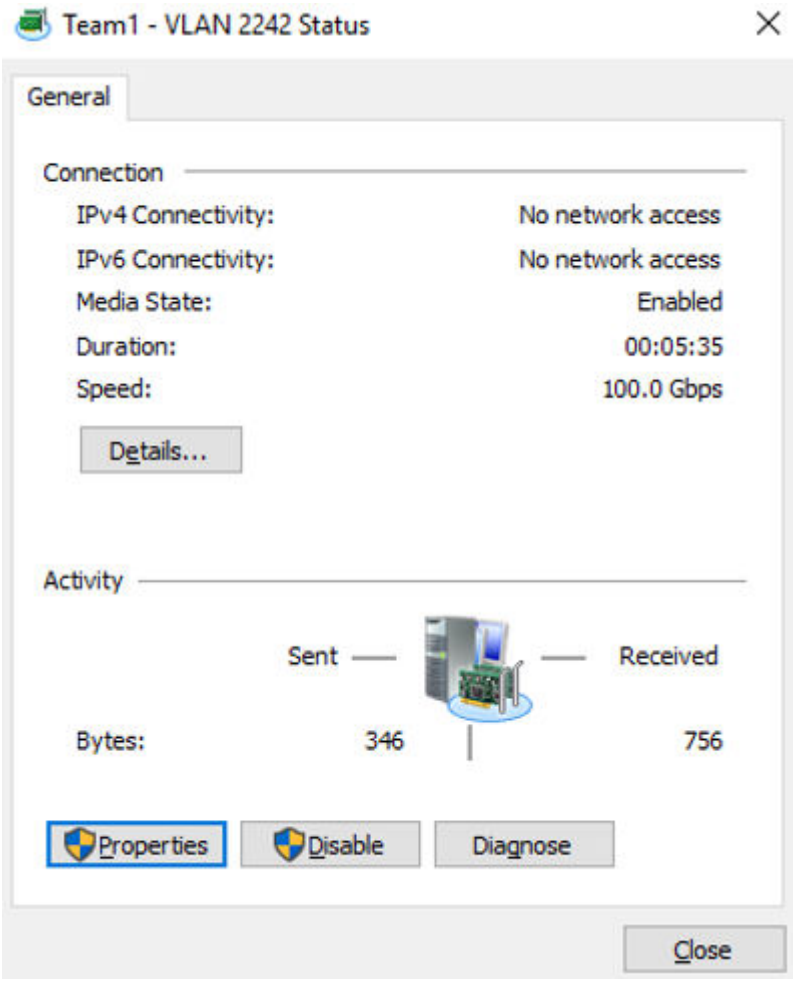
g. On the **Team1 - VLAN 2242 Properties** page, click **OK** to save the settings.

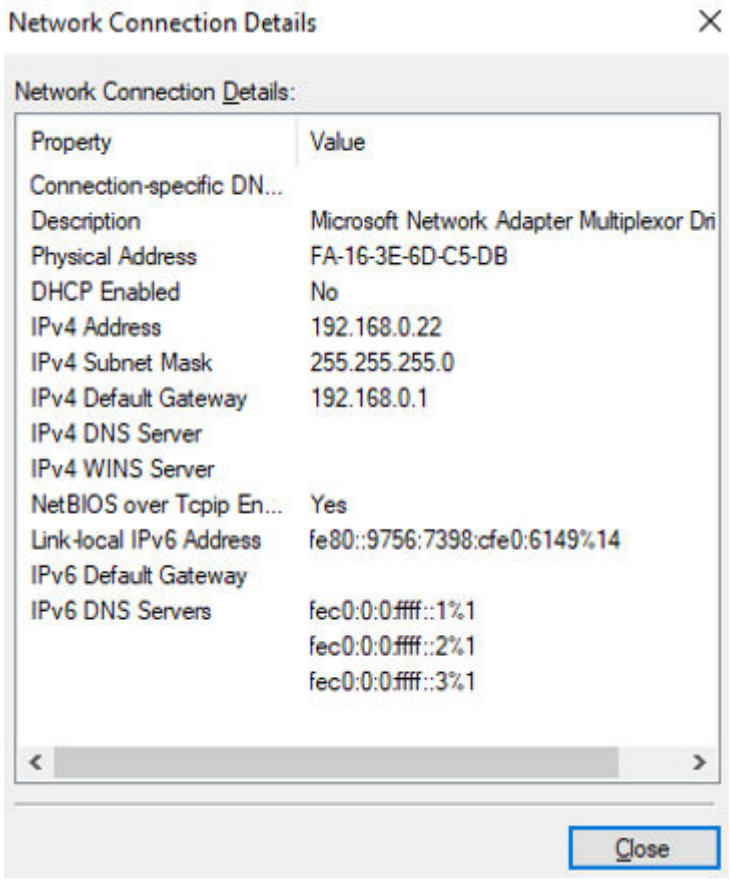


h. Return to the **Team1 - VLAN 2242 Status** page and click **Details...**.

On the **Network Connection Details** page, check whether the following information is correctly configured:

- **Physical Address**: MAC address of the supplementary network interface.
- **IPv4 Address**: the private IP address of the supplementary network interface.
- **IPv4 Subnet Mask**: the mask of the subnet where the supplementary network interface is created.
- **IPv4 Default Gateway**: the gateway of the subnet where the supplementary network interface is created.





i. Check the settings and click **Close**.

8. On the Windows PowerShell CLI page, check whether the network interface and supplementary network interface are connected to the test ECS.

   a. Run the following command to verify the connectivity between network interface **eth0** and the test ECS:

      Ping *<private-IP-address-of-the-test-ECS>* **-S** *<private-IP-address-of-the-network-interface>*

      Plan the same VPC and security group for the test ECS and the ECS with network interface **eth0** attached. This allows the two ECSs to communicate with each other by default.

      Example command:

      Ping 192.168.0.133 -S 192.168.0.16

      If information similar to the following is displayed, the two ECSs can communicate with each other.

```
PS C:\Users\Administrator> Ping 192.168.0.133 -S 192.168.0.16

Pinging 192.168.0.133 from 192.168.0.16 with 32 bytes of data:
Reply from 192.168.0.133: bytes=32 time=1ms TTL=64
Reply from 192.168.0.133: bytes=32 time<1ms TTL=64
Reply from 192.168.0.133: bytes=32 time<1ms TTL=64
Reply from 192.168.0.133: bytes=32 time<1ms TTL=64

Ping statistics for 192.168.0.133:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 1ms, Average = 0ms
```


b. Run the following command to verify the connectivity between the supplementary network interface of **eth0** and the test ECS:

Ping *<private-IP-address-of-the-test-ECS>* **-S** *<private-IP-address-of-the-supplementary-network-interface>*

Plan the same VPC and security group for the test ECS and the ECS with the supplementary network interface attached. This allows the two ECSs to communicate with each other by default.

Example command:

Ping 192.168.0.133 -S 192.168.0.22

If information similar to the following is displayed, the two ECSs can communicate with each other.

```
PS C:\Users\Administrator> Ping 192.168.0.133 -S 192.168.0.22

Pinging 192.168.0.133 from 192.168.0.22 with 32 bytes of data:
Reply from 192.168.0.133: bytes=32 time=1ms TTL=64
Reply from 192.168.0.133: bytes=32 time<1ms TTL=64
Reply from 192.168.0.133: bytes=32 time<1ms TTL=64
Reply from 192.168.0.133: bytes=32 time<1ms TTL=64

Ping statistics for 192.168.0.133:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 1ms, Average = 0ms
```

5.2.3 Viewing the Basic Information About a Supplementary Network Interface

Scenarios

You can view basic information about your supplementary network interface on the management console, including its ID, network interface, VLAN ID, VPC, subnet, private IP address, EIP, MAC address, and security groups.

Procedure

1. Log in to the management console.
2. Click in the upper left corner and select the desired region and project.
3. Click in the upper left corner and choose **Networking** > **Virtual Private Cloud**.

   The **Virtual Private Cloud** page is displayed.
4. In the navigation pane on the left, choose **Virtual Private Cloud** > **Network Interfaces**.
5. On the **Network Interfaces** page, click **Supplementary Network Interfaces** tab.
6. Click the private IP address of the supplementary network interface whose details you want to view.
   - On the **Summary** tab, you can view its ID, network interface, VLAN ID, VPC, subnet, private IP address, EIP, and MAC address.
   - On the **Associated Security Groups** tab, you can view its associated security groups and their rules.

Other Operations

On the supplementary network interface details page, you can also modify the following information:

- On the **Summary** tab, you can modify the description of the interface and change its bound EIP.
- On the **Associated Security Groups** tab, you can change the associated security groups of the interface. For details, see **Changing Security Groups That Are Associated with a Supplementary Network Interface**.

5.2.4 Binding or Unbinding an EIP to or from a Supplementary Network Interface

Scenarios

You can bind a supplementary network interface to an EIP.

A supplementary network interface has a private IP address. You can also bind an EIP to the interface. The binding between a supplementary network interface and an EIP does not change when the network interface of the supplementary network interface is detached from an ECS and then attached to another ECS. The supplementary network interface still has its private IP address and EIP.

A network interface can have multiple supplementary network interfaces attached. If each supplementary network interface has an EIP, the ECS with the network interface attached can have multiple EIPs for flexible Internet access.

If you do not need an EIP or want to delete a supplementary network interface, you can unbind the EIP from the interface.

Binding a Supplementary Network Interface to an EIP

1. Log in to the management console.
2. Click in the upper left corner and select the desired region and project.
3. Click in the upper left corner and choose **Networking** > **Virtual Private Cloud**.

   The **Virtual Private Cloud** page is displayed.
4. In the navigation pane on the left, choose **Virtual Private Cloud** > **Network Interfaces**.
5. On the **Network Interfaces** page, click **Supplementary Network Interfaces** tab.
6. Locate the row that contains the supplementary network interface, click **Bind EIP** in the **Operation** column, and select the EIP to be bound.
7. Click **OK**.

Unbinding a Supplementary Network Interface from an EIP

1. Log in to the management console.

2. Click in the upper left corner and select the desired region and project.
3. Click in the upper left corner and choose **Networking** > **Virtual Private Cloud**.

   The **Virtual Private Cloud** page is displayed.
4. In the navigation pane on the left, choose **Virtual Private Cloud** > **Network Interfaces**.
5. On the **Network Interfaces** page, click **Supplementary Network Interfaces** tab.
6. Locate the row that contains the supplementary network interface and click **Unbind EIP** in the **Operation** column.
7. Click **OK**.

5.2.5 Changing Security Groups That Are Associated with a Supplementary Network Interface

Scenarios

After a supplementary network interface is created, you can change its security group.

You can change the security group of a supplementary network interface:

- On the page of the supplementary network interface list.
- On the details page of a supplementary network interface.

Procedure

Changing the security group associated with a supplementary network interface on the supplementary network interface list page

1. Log in to the management console.
2. Click in the upper left corner and select the desired region and project.
3. Click in the upper left corner and choose **Networking** > **Virtual Private Cloud**.

   The **Virtual Private Cloud** page is displayed.
4. In the navigation pane on the left, choose **Virtual Private Cloud** > **Network Interfaces**.
5. On the **Network Interfaces** page, click the **Supplementary Network Interfaces** tab.
6. Locate the row that contains the supplementary network interface and click **Change Security Group** in the **Operation** column.
7. On the **Change Security Group** page, select the security group to be associated.
8. Click **OK**.

Changing the security group associated with a supplementary network interface on the supplementary network interface details page

1. Log in to the management console.
2. Click in the upper left corner and select the desired region and project.
3. Click in the upper left corner and choose **Networking** > **Virtual Private Cloud**.

   The **Virtual Private Cloud** page is displayed.
4. In the navigation pane on the left, choose **Virtual Private Cloud** > **Network Interfaces**.
5. On the **Network Interfaces** page, click the **Supplementary Network Interfaces** tab.
6. Click the private IP address of the supplementary network interface whose security group is to be changed.
7. Click the **Associated Security Groups** tab. Then, click **Change Security Group**.
8. On the **Change Security Group** page, select the security group to be associated.
9. Click **OK**.

5.2.6 Deleting a Supplementary Network Interface

Scenarios

If you want to delete a supplementary network interface with an EIP bound, you have to first unbind the EIP from the interface.

Notes and Constraints

- Deleting a supplementary network interface will also detach it from its network interface.
- Deleting a supplementary network interface will also unbind its EIP. You can choose to release the EIP if needed.

  If you do not release the EIP, the EIP will continue to be billed, but you can still bind it to other cloud resources.
- If a supplementary network interface has resources associated, deleting the interface will also delete any rules for associated resources that reference it.

  For example, deleting a supplementary network interface that is used as the next hop for a custom route in a VPC route table will also delete the associated route.

Procedure

1. Log in to the management console.
2. Click in the upper left corner and select the desired region and project.
3. Click in the upper left corner and choose **Networking** > **Virtual Private Cloud**.

   The **Virtual Private Cloud** page is displayed.
4. In the navigation pane on the left, choose **Virtual Private Cloud** > **Network Interfaces**.

5. On the **Network Interfaces** page, click **Supplementary Network Interfaces** tab.
6. Locate the row that contains the supplementary network interface and click **Delete** in the **Operation** column.

   A confirmation dialog box is displayed.
7. Confirm the information and click **OK**.

   Deleting a supplementary network interface will also delete the VLAN sub-interfaces configured on the ECS.

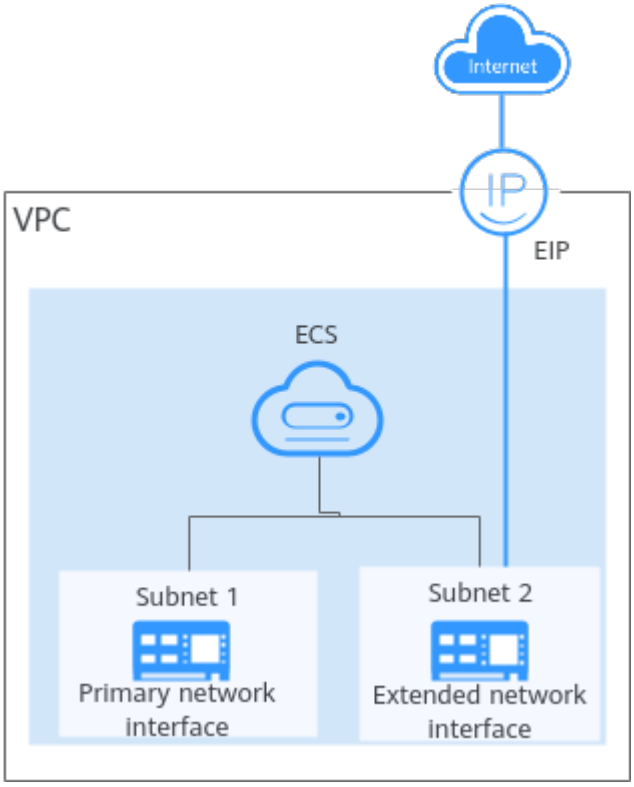
5.3 Network Interface Configuration Examples

5.3.1 Binding an EIP to the Extended Network Interface of an ECS to Enable Internet Access

Scenarios

As shown in **Figure 5-2**, the ECS has two network interfaces, one primary network interface and one extended network interface. You can bind an EIP to the extended network interface of the ECS and configure policy-based routes to ensure that the ECS can access the Internet through the EIP.

Figure 5-2 Accessing the Internet through the EIP bound to the extended network interface



NOTE

This section uses a Linux ECS as an example.

Step 1: Create Cloud Resources and Attach an Extended Network Interface

1. Create a VPC and two subnets in the VPC.

   In this example, the primary and extended network interfaces of the ECS are in different subnets.

   For details, see **Creating a VPC and Subnet**.
2. Create an ECS in the VPC subnet.

   For details, see **Purchasing a Custom ECS**.
3. Create a network interface and attach it to the ECS as an extended network interface.

   When creating a network interface, select a different subnet from where the primary network interface is created. For details, see **Creating a Network Interface**.

   Attach the network interface to the ECS. For details, see **Attaching a Network Interface to a Cloud Server**.
4. Assign an EIP and bind it to the extended network interface of the ECS.

   For details, see **Assigning an EIP**.

   Bind the EIP to the extended network interface of the ECS. For details, see **Binding an EIP to a Network Interface**.

Step 2: Obtain the ECS Network Information

Before configuring policy-based routes for the extended network interface, you need to obtain the network information in **Table 5-7**.

Table 5-7 Required ECS network information

Item	Primary Network Interface	Extended Network Interface
Private IP address of the network interface	192.168.11.42	192.168.17.191
Subnet gateway address	192.168.11.1	192.168.17.1

1. Obtain the private IP addresses of the ECS's network interfaces.
   a. Log in to the management console.
   b. Click  in the upper left corner and select the desired region and project.
   c. Click **Service List** and choose **Compute** > **Elastic Cloud Server**.

d. In the ECS list, locate the target ECS and click its name.

The **Summary** tab page of the ECS is displayed.

e. Click the **Network Interfaces** tab and view the private IP addresses of the primary and extended network interfaces of the ECS.

2. Obtain the gateway address of the subnet.

a. Log in to the management console.

b. Click in the upper left corner and select the desired region and project.

c. Click **Service List** and choose **Compute** > **Elastic Cloud Server**.

d. In the ECS list, locate the target ECS and click its name.

The **Summary** tab page of the ECS is displayed.

e. In the **ECS Information** area, click the VPC name.

The **Virtual Private Cloud** page is displayed.

f. In the VPC list and click the number in the **Subnets** column.

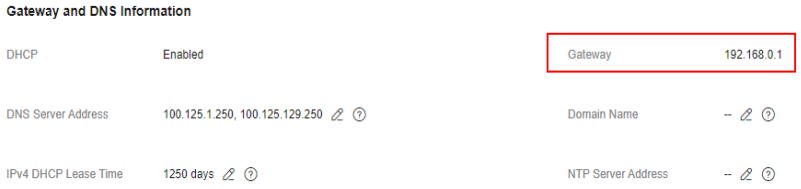
The **Subnets** page is displayed.

g. In the subnet list, click the subnet name.

The **Summary** page is displayed.

h. In the **Gateway and DNS Information** area, view the gateway address of the subnet.

Figure 5-3 Viewing the gateway address of the subnet



Step 3: Configure Policy-based Routes for the Extended Network Interface

1. ECS Remotely log in to the ECS.

For details, see **How Do I Log In to My ECS?**

2. Run the following command to query the route information of the network interface:

route -n

The following figure is displayed. In this figure:

- The destination of the route for the primary network interface is 192.168.11.0/24.
- The destination of the route for the extended network interface is 192.168.17.0/24.


```
[root@ecs-b926 ~]# route -n
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
0.0.0.0         192.168.11.1    0.0.0.0         UG    0      0        0 eth0
169.254.0.0     0.0.0.0         255.255.0.0     U     1002   0        0 eth0
169.254.0.0     0.0.0.0         255.255.0.0     U     1003   0        0 eth1
169.254.169.254 192.168.11.1    255.255.255.255 UGH   0      0        0 eth0
192.168.11.0    0.0.0.0         255.255.255.0   U     0      0        0 eth0
192.168.17.0    0.0.0.0         255.255.255.0   U     0      0        0 eth1
[root@ecs-b926 ~]#
```

3. Run the following command to query the network interface names of the ECS:

   ifconfig

   The following figure is displayed. Search for the network interface name based on the network interface address. In this figure:

   - 192.168.11.42 is the IP address of the primary network interface, and the network interface name is eth0.
   - 192.168.17.191 is the IP address of the extended network interface, and the network interface name is eth1.

```
[root@ecs-b926~]# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.11.42  netmask 255.255.255.0  broadcast 192.168.11.255
        inet6 fe80::f816:3eff:fef7:1c44  prefixlen 64  scopeid 0x20<link>
        ether fa:16:3e:f7:1c:44  txqueuelen 1000  (Ethernet)
        RX packets 127  bytes 21633 (21.1 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 258  bytes 22412 (21.8 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

eth1: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.17.191  netmask 255.255.255.0  broadcast 192.168.17.255
        inet6 fe80::f816:3eff:fe1c:b57f  prefixlen 64  scopeid 0x20<link>
        ether fa:16:3e:1c:b5:7f  txqueuelen 1000  (Ethernet)
        RX packets 11  bytes 1283 (1.2 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 12  bytes 1388 (1.3 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1  (Local Loopback)
        RX packets 51  bytes 12018 (11.7 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 51  bytes 12018 (11.7 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

4. Configure the default route for the ECS so that it can access the Internet through the extended network interface.

   a. Run the following command to delete the default route of the primary network interface:

      route del -net 0.0.0.0 gw *<subnet-gateway-IP-address>* **dev** *<network interface-name>*

      The parameters are described as follows:

      - 0.0.0.0: destination IP address, indicating that multiple IP addresses are matched. Do not change the value.
      - Subnet gateway IP address: Enter the subnet gateway address of the primary network interface collected in section **Table 5-7**.

- Network interface name: Enter the name of the primary network interface obtained in **3**.

Example command:

route del -net 0.0.0.0 gw 192.168.11.1 dev eth0

NOTE

This operation will interrupt ECS traffic.

b. Run the following command to configure the default route for the extended network interface:

route add default gw *Subnet-gateway-IP-address*

The parameters are described as follows:

Subnet gateway IP address: Enter the subnet gateway address of the extended network interface collected in section **Table 5-7**.

Example command:

route add default gw 192.168.17.1

5. Verify network connectivity.

Run the following command to check whether the ECS can access the Internet:

ping *Public-IP-address-or-domain-name*

Example command:

ping support.huaweicloud.com

If information similar to the following is displayed, the ECS can communicate with the Internet.

```
[root@ecs-a01 ~]# ping support.huaweicloud.com
PING hcdnw.cbg-notzj.c.cdnhwc2.com (203.193.226.103) 56(84) bytes of data.
64 bytes from 203.193.226.103 (203.193.226.103): icmp_seq=1 ttl=51 time=2.17 ms
64 bytes from 203.193.226.103 (203.193.226.103): icmp_seq=2 ttl=51 time=2.13 ms
64 bytes from 203.193.226.103 (203.193.226.103): icmp_seq=3 ttl=51 time=2.10 ms
64 bytes from 203.193.226.103 (203.193.226.103): icmp_seq=4 ttl=51 time=2.09 ms
...
--- hcdnw.cbg-notzj.c.cdnhwc2.com ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3004ms
rtt min/avg/max/mdev = 2.092/2.119/2.165/0.063 ms
```

5.3.2 Configuring Policy-based Routes for an ECS with Multiple Network Interfaces

5.3.2.1 Overview

Background

If a cloud server has multiple network interfaces, the primary network interface can communicate with external networks by default, but the extended network interfaces cannot. To enable extended network interfaces to communicate with external networks, you need to configure policy-based routes for these network interfaces.

Scenarios

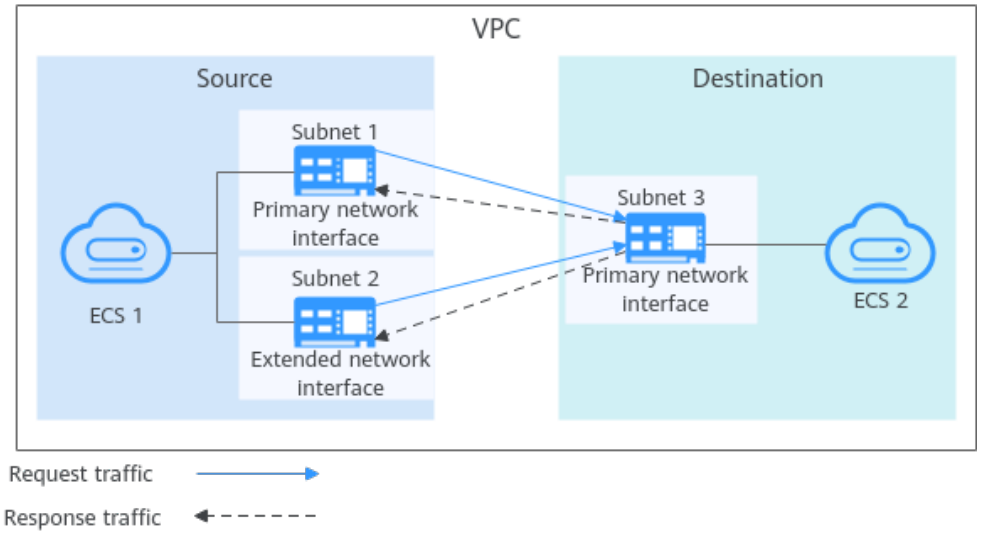
This example describes how to configure policy-based routes for an ECS with two network interfaces. **Figure 5-4** shows the networking.

- The primary and extended network interfaces on the source ECS are in different subnets of the same VPC.
- The source and destination ECSs are in different subnets of the same VPC and the two ECSs can communicate with each other through primary network interfaces without configuring policy-based routes.
- After policy-based routes are configured for the two network interfaces of the source ECS, both the primary and extended network interfaces can be used to communicate with the destination ECS.

NOTICE

You can select a destination IP address based on service requirements. Before configuring policy-based routes, ensure that the source ECS can use its primary network interface to communicate with the destination ECS.

Figure 5-4 Networking of an ECS with two network interfaces



Operation Guide

You can follow the following operations to configure policy-based routes for Linux and Windows ECSs. For details, see **Table 5-8**.

Table 5-8 Operation instructions

OS	IP Address Version	Description
Linux	IPv4	An ECS running CentOS 8.0 (64-bit): **Configuring IPv4 and IPv6 Policy-based Routes for a Linux ECS with Multiple Network Interfaces (CentOS)** An ECS running Ubuntu 22.04 server (64-bit): **Configuring IPv4 and IPv6 Policy-based Routes for a Linux ECS with Multiple Network Interfaces (Ubuntu)**
	IPv6	
Windows	IPv4	An ECS running Windows Server 2012 (64-bit): **Configuring IPv4 and IPv6 Policy-based Routes for a Windows ECS with Multiple Network Interfaces**
	IPv6	

5.3.2.2 Collecting ECS Network Information

Scenarios

Before configuring policy-based routes for an ECS with multiple network interfaces, you need to collect network information about the ECS.

- **Table 5-9** lists the information to be collected for a Linux ECS using IPv4.

  Table 5-9 Linux ECS using IPv4

ECS	Primary Network Interface	Extended Network Interface	How to Obtain
Source	• IP address: 10.0.0.115 • Subnet: 10.0.0.0/24 • Subnet gateway: 10.0.0.1	• IP address: 10.0.1.183 • Subnet: 10.0.1.0/24 • Subnet gateway: 10.0.1.1	• **Obtaining ECS Network Interface Addresses** • **Obtaining Subnet CIDR Blocks and Gateway Addresses**
Destination	IP address: 10.0.2.12	N/A	

- **Table 5-10** lists the information to be collected for a Linux ECS using IPv6.

Table 5-10 Linux ECS using IPv6

ECS	Primary Network Interface	Extended Network Interface	How to Obtain
Source	• IPv4 address: 10.0.0.102 • IPv6 address: 2407:c080:1200:1dd8:859c:e5d5:8b3d:a2d9 • IPv6 subnet: 2407:c080:1200:1dd8::/64 • IPv6 subnet gateway: 2407:c080:1200:1dd8::1	• IPv4 address: 10.0.1.191 • IPv6 address: 2407:c080:1200:1a9c:7cc0:63b5:8e65:4dd8 • IPv6 subnet: 2407:c080:1200:1a9c::/64 • IPv6 subnet gateway: 2407:c080:1200:1a9c::1	• **Obtaining ECS Network Interface Addresses** • **Obtaining Subnet CIDR Blocks and Gateway Addresses**
Destination	• IPv4 address: 10.0.2.3 • IPv6 address: 2407:c080:1200:1dd9:16a7:fe7a:8f71:7044	N/A	

- **Table 5-11** lists the information to be collected for a Windows ECS using IPv4.

Table 5-11 Windows ECS using IPv4

ECS	Primary Network Interface	Extended Network Interface	How to Obtain
Source	• IP address: 10.0.0.59 • Subnet gateway: 10.0.0.1	• IP address: 10.0.1.104 • Subnet gateway: 10.0.1.1	• **Obtaining ECS Network Interface Addresses** • **Obtaining Subnet CIDR Blocks and Gateway Addresses**
Destination	IP address: 10.0.2.12	N/A	

- **Table 5-12** lists the information to be collected for a Windows ECS using IPv6.

Table 5-12 Windows ECS using IPv6

ECS	Primary Network Interface	Extended Network Interface	How to Obtain
Source	IP address: 2407:c080:802:aba:6788:fb94:d71f:8deb	IP address: 2407:c080:802:be6:71c8:42e0:d44e:eeb4	Obtaining ECS Network Interface Addresses
Destination	IP address: 2407:c080:802:be7:c2e6:d99c:b685:c6c8	N/A	

NOTICE

The above information is only for your reference.

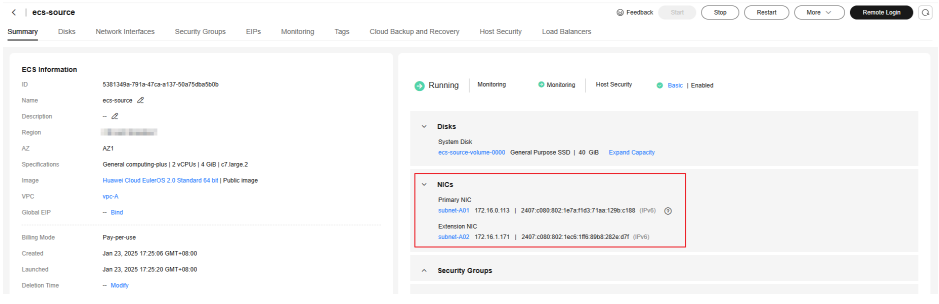
Obtaining ECS Network Interface Addresses

1. Log in to the management console.
2. Click in the upper left corner and select the desired region and project.
3. Click **Service List** and choose **Compute** > **Elastic Cloud Server**.
4. In the ECS list, click the target ECS name.

   The **Summary** tab page of the ECS is displayed.
5. In the **NICs** area, view the IP addresses of the primary and extended network interfaces.

   You can view the IPv4 and IPv6 addresses of network interfaces.

Figure 5-5 IPv4 and IPv6 addresses of network interfaces

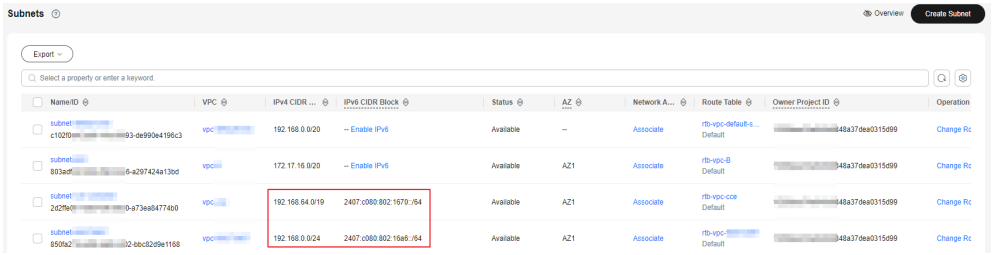


Obtaining Subnet CIDR Blocks and Gateway Addresses

1. Log in to the management console.
2. Click in the upper left corner and select the desired region and project.
3. Click **Service List** and choose **Compute** > **Elastic Cloud Server**.

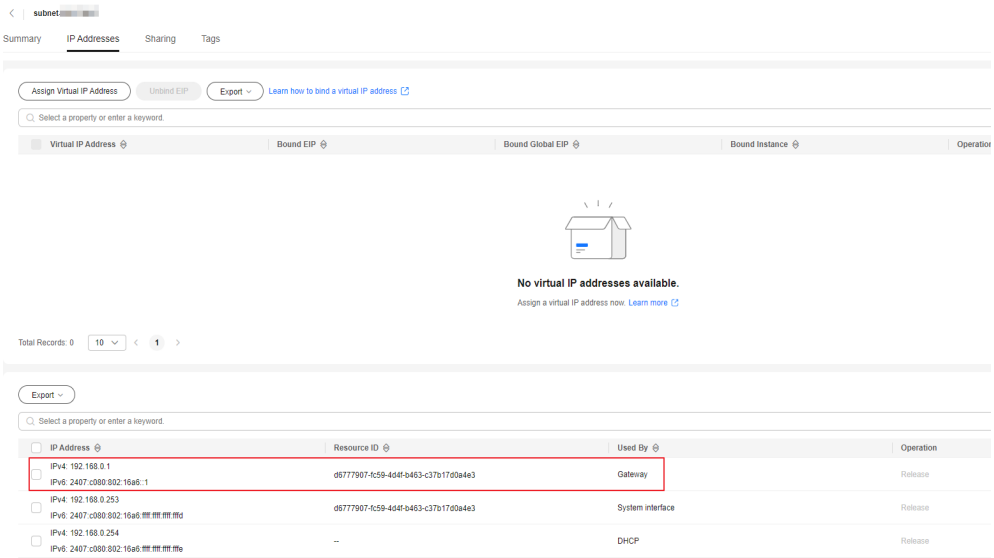
4. In the ECS list, click the target ECS name.
   The **Summary** tab page of the ECS is displayed.
5. In the **ECS Information** area, click the name of its VPC.
   The **Virtual Private Cloud** page is displayed.
6. Locate the target VPC and click the number in the **Subnets** column.
   The **Subnets** page is displayed.
7. In the subnet list, view the CIDR blocks of the subnets.
   You can view the IPv4 and IPv6 CIDR blocks of subnets.

Figure 5-6 IPv4 and IPv6 CIDR blocks of subnets



8. In the subnet list, click the subnet name.
   The **Summary** page is displayed.
9. Click the **IP Addresses** tab and view the gateway addresses of the subnet.
   You can view the IPv4 and IPv6 addresses of a gateway.

Figure 5-7 IPv4 and IPv6 addresses of a gateway



5.3.2.3 Configuring IPv4 and IPv6 Policy-based Routes for a Linux ECS with Multiple Network Interfaces (CentOS)

Scenarios

This section describes how to configure policy-based routes for a CentOS 8.0 64-bit ECS with two network interfaces.

- IPv4: **Configuring IPv4 Policy-based Routes for a CentOS ECS**
- IPv6: **Configuring IPv6 Policy-based Routes for a CentOS ECS**

For details about the background knowledge and networking of an ECS with two network interfaces, see **Overview**.

Configuring IPv4 Policy-based Routes for a CentOS ECS

1. Collect the ECS network interface information required for configuring policy-based routes.

   For details, see **Collecting ECS Network Information**.

   In this example, the network information of the ECS is shown in **Table 5-13**.

   Table 5-13 CentOS ECS using IPv4

ECS	Primary Network Interface	Extended Network Interface
Source	• IP address: 10.0.0.115 • Subnet: 10.0.0.0/24 • Subnet gateway: 10.0.0.1	• IP address: 10.0.1.183 • Subnet: 10.0.1.0/24 • Subnet gateway: 10.0.1.1
Destination	IP address: 10.0.2.12	N/A

2. Log in to the source ECS.

   For details, see **How Do I Log In to My ECS?**

3. Check whether the source ECS can use its primary network interface to communicate with the destination ECS:

   Before configuring policy-based routes, ensure that the source ECS can use its primary network interface to communicate with the destination ECS.

   ping -I *<IP-address-of-the-primary-network-interface-on-the-source-ECS> <IP-address-of-the-destination-ECS>*

   In this example, run the following command:

   ping -I 10.0.0.115 10.0.2.12

   If information similar to the following is displayed, the source ECS can use its primary network interface to communicate with the destination ECS.

   ```
   [root@ecs-resource ~]# ping -I 10.0.0.115 10.0.2.12
   PING 10.0.2.12 (10.0.2.12) from 10.0.0.115 : 56(84) bytes of data.
   64 bytes from 10.0.2.12: icmp_seq=1 ttl=64 time=0.775 ms
   64 bytes from 10.0.2.12: icmp_seq=2 ttl=64 time=0.268 ms
   64 bytes from 10.0.2.12: icmp_seq=3 ttl=64 time=0.220 ms
   64 bytes from 10.0.2.12: icmp_seq=4 ttl=64 time=0.167 ms
   ^C
   --- 10.0.2.12 ping statistics ---
   ```

4. Query the network interface names of the source ECS:

   ifconfig

   Search for the network interface names based on IP addresses.

   – The primary network interface address is 10.0.0.115, and its name is eth0.

– The extended network interface address is 10.0.1.183, and its name is eth1.

```
[root@ecs-resource ~]# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
      inet 10.0.0.115  netmask 255.255.255.0  broadcast 10.0.0.255
      inet6 fe80::f816:3eff:fe92:6e0e  prefixlen 64  scopeid 0x20<link>
      ether fa:16:3e:92:6e:0e  txqueuelen 1000  (Ethernet)
      RX packets 432288  bytes 135762012 (129.4 MiB)
      RX errors 0  dropped 0  overruns 0  frame 1655
      TX packets 423744  bytes 106716932 (101.7 MiB)
      TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

eth1: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
      inet 10.0.1.183  netmask 255.255.255.0  broadcast 10.0.1.255
      inet6 fe80::f816:3eff:febf:5818  prefixlen 64  scopeid 0x20<link>
      ether fa:16:3e:bf:58:18  txqueuelen 1000  (Ethernet)
      RX packets 9028  bytes 536972 (524.3 KiB)
      RX errors 0  dropped 0  overruns 0  frame 1915
      TX packets 6290  bytes 272473 (266.0 KiB)
      TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

5. Configure temporary routes for the source ECS.

NOTICE

Temporary routes are applied immediately but are lost after ECS restarts. To avoid network disruptions, perform **6** to configure permanent routes instead.

a. Configure policy-based routes for both the primary and extended network interfaces.

- Primary network interface

  ip route add default via *<subnet-gateway>* **dev** *<network-interface-name>* **table** *<route-table-name>*

  ip route add *<subnet-CIDR-block>* **dev** *<network-interface-name>* **table** *<route-table-name>*

  ip rule add from *<network-interface-address>* **table** *<route-table-name>*

- Extended network interface

  ip route add default via *<subnet-gateway>* **dev** *<network-interface-name>* **table** *<route-table-name>*

  ip route add *<subnet-CIDR-block>* **dev** *<network-interface-name>* **table** *<route-table-name>*

  ip rule add from *<network-interface-address>* **table** *<route-table-name>*

  Configure the parameters as follows:

- Network interface name: Enter the name obtained in **4**.
- Route table name: Name the route table with a number.
- Other network information: Enter the IP addresses collected in **1**.

  In this example, run the following commands:

- Primary network interface

  ip route add default via 10.0.0.1 dev eth0 table 10

  ip route add 10.0.0.0/24 dev eth0 table 10

  ip rule add from 10.0.0.115 table 10

- Extended network interface

  ip route add default via 10.0.1.1 dev eth1 table 20

  ip route add 10.0.1.0/24 dev eth1 table 20

  ip rule add from 10.0.1.183 table 20

NOTE

If the ECS has multiple network interfaces, configure policy-based routes for all network interfaces one by one.

b. Check whether the policy-based routes are added.

ip rule

ip route show table *<route-table-name-of-the-primary-network-interface>*

ip route show table *<route-table-name-of-the-extended-network-interface>*

The route table name is the one configured in **5.a**.

In this example, run the following commands:

ip rule

ip route show table 10

ip route show table 20

If information similar to the following is displayed, the policy-based routes have been added.

```
[root@ecs-resource ~]# ip rule
0:      from all lookup local
32764:  from 10.0.1.183 lookup 20
32765:  from 10.0.0.115 lookup 10
32766:  from all lookup main
32767:  from all lookup default
[root@ecs-resource ~]# ip route show table 10
default via 10.0.0.1 dev eth0
10.0.0.0/24 dev eth0 scope link
[root@ecs-resource ~]# ip route show table 20
default via 10.0.1.1 dev eth1
10.0.1.0/24 dev eth1 scope link
```

c. Check whether the source and destination ECSs can communicate with each other.

ping -I *<IP-address-of-the-primary-network-interface-on-the-source-ECS> <IP-address-of-the-destination-ECS>*

ping -I *<IP-address-of-the-extended-network-interface-on-the-source-ECS> <IP-address-of-the-destination-ECS>*

In this example, run the following commands:

ping -I 10.0.0.115 10.0.2.12

ping -I 10.0.1.183 10.0.2.12

If information similar to the following is displayed, both the network interfaces of the source ECS can communicate with the destination ECS.


```
[root@ecs-resource ~]# ping -I 10.0.0.115 10.0.2.12
PING 10.0.2.12 (10.0.2.12) from 10.0.0.115 : 56(84) bytes of data.
64 bytes from 10.0.2.12: icmp_seq=1 ttl=64 time=0.775 ms
64 bytes from 10.0.2.12: icmp_seq=2 ttl=64 time=0.268 ms
64 bytes from 10.0.2.12: icmp_seq=3 ttl=64 time=0.220 ms
64 bytes from 10.0.2.12: icmp_seq=4 ttl=64 time=0.167 ms
^C
--- 10.0.2.12 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 102ms
rtt min/avg/max/mdev = 0.167/0.357/0.775/0.244 ms
[root@ecs-resource ~]# ping -I 10.0.1.183 10.0.2.12
PING 10.0.2.12 (10.0.2.12) from 10.0.1.183 : 56(84) bytes of data.
64 bytes from 10.0.2.12: icmp_seq=1 ttl=64 time=2.84 ms
64 bytes from 10.0.2.12: icmp_seq=2 ttl=64 time=0.258 ms
64 bytes from 10.0.2.12: icmp_seq=3 ttl=64 time=0.234 ms
64 bytes from 10.0.2.12: icmp_seq=4 ttl=64 time=0.153 ms
^C
--- 10.0.2.12 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 92ms
rtt min/avg/max/mdev = 0.153/0.871/2.840/1.137 ms
```

6. Configure permanent routes for the source ECS.

   a. Run the following command to open the **/etc/rc.local** file:

      vi /etc/rc.local

   b. Press **i** to enter the editing mode.

   c. Add the following content to the end of the file:

```
# check eth0
for ((x=0; x<10; x++)); do
  if (ip addr show eth0 | grep -w 10.0.0.115 >/dev/null 2>&1); then
    break
  fi
  sleep 1
done

# Add v4 routes for eth0
ip route flush table 10
ip route add default via 10.0.0.1 dev eth0 table 10
ip route add 10.0.0.0/24 dev eth0 table 10
ip rule add from 10.0.0.115 table 10

# check eth1
for ((x=0; x<10; x++)); do
  if (ip addr show eth1 | grep -w 10.0.1.183 >/dev/null 2>&1); then
    break
  fi
  sleep 1
done

# Add v4 routes for eth1
ip route flush table 20
ip route add default via 10.0.1.1 dev eth1 table 20
ip route add 10.0.1.0/24 dev eth1 table 20
ip rule add from 10.0.1.183 table 20
# Add v4 routes for cloud-init
ip rule add to 169.254.169.254 table main
```

   The parameters are as follows:

   - **check eth0**: Check for the presence of the IP address 10.0.0.115 on primary network interface eth0 every second up to 10 times.
   - **Add v4 routes for eth0**: Add policy-based routes for the primary network interface. Set the value to be the same as that configured in **5.a**.

- **check eth1**: Check for the presence of the IP address 10.0.1.183 on extended network interface eth1 every second up to 10 times.
- **Add v4 routes for eth1**: Add policy-based routes for the extended network interface. Set the value to be the same as that configured in 5.a.
- **Add v4 routes for cloud-init**: Set the Cloud-Init address to the same value as that in this example.

d. Press **ESC** to exit and enter **:wq!** to save the configuration.

e. Run the following command to add execute permissions to the **/etc/rc.local** file:

chmod +x /etc/rc.local

NOTE

If your operating system is Red Hat or EulerOS, run the following command after you perform 6.e:

chmod +x /etc/rc.d/rc.local

f. Run the following command to restart the ECS:

reboot

NOTICE

Policy-based routes added to the **/etc/rc.local** file take effect only after the ECS is restarted. Ensure that workloads on the ECS will not be affected before restarting the ECS.

g. Repeat 5.b to 5.c to check whether the policy-based routes are added and whether the source ECS and the destination ECS can communicate with each other.

Configuring IPv6 Policy-based Routes for a CentOS ECS

1. Collect the ECS network interface information required for configuring policy-based routes.

   For details, see **Collecting ECS Network Information**.

Table 5-14 CentOS ECS using IPv6

ECS	Primary Network Interface	Extended Network Interface
Source	• IPv4 address: 10.0.0.102 • IPv6 address: 2407:c080:1200:1dd8:859c:e5d5:8b3d:a2d9 • IPv6 subnet: 2407:c080:1200:1dd8::/64 • IPv6 subnet gateway: 2407:c080:1200:1dd8::1	• IPv4 address: 10.0.1.191 • IPv6 address: 2407:c080:1200:1a9c:7cc0:63b5:8e65:4dd8 • IPv6 subnet: 2407:c080:1200:1a9c::/64 • IPv6 subnet gateway: 2407:c080:1200:1a9c::1

ECS	Primary Network Interface	Extended Network Interface
Destination	• IPv4 address: 10.0.2.3 • IPv6 address: 2407:c080:1200:1dd9:16a7:fe7a:8f71:7044	N/A

2. Log in to the source ECS.

   For details, see **How Do I Log In to My ECS?**

3. Check whether the ECSs have IPv6 enabled and have IPv6 addresses.

   NOTICE

   Perform this step for both the source and destination ECSs to ensure that the ECSs have IPv6 addresses. Otherwise, the ECSs cannot communicate with each other using IPv6 addresses.

   ECSs in this example run CentOS 8.0 (64-bit). For details about how to assign IPv6 addresses for ECSs running other OSs, see **Dynamically Assigning IPv6 Addresses**.

   a. Run the following command to check whether the ECS has IPv6 addresses:

      ip addr

      In the following command output, eth0 and eth1 are the network interfaces of the ECS. Each network interface has one **inet6** entry starting with **fe80**. This indicates that the ECS has IPv6 enabled but has not been assigned IPv6 addresses. In this case, perform **3.b** to **3.g** to assign IPv6 addresses.

      ```
      [root@ecs-resource ~]# ip addr
      ...
      2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
          link/ether fa:16:3e:22:22:88 brd ff:ff:ff:ff:ff:ff
          inet 10.0.0.102/24 brd 10.0.0.255 scope global dynamic noprefixroute eth0
             valid_lft 107943256sec preferred_lft 107943256sec
          inet6 fe80::f816:3eff:fe22:2288/64 scope link
             valid_lft forever preferred_lft forever
      3: eth1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
          link/ether fa:16:3e:22:23:e1 brd ff:ff:ff:ff:ff:ff
          inet 10.0.1.191/24 brd 10.0.1.255 scope global dynamic noprefixroute eth1
             valid_lft 107943256sec preferred_lft 107943256sec
          inet6 fe80::f816:3eff:fe22:23e1/64 scope link
             valid_lft forever preferred_lft forever
      ```

   b. Query the network interface names of the source ECS:

      ifconfig

      Search for the network interface names based on IP addresses.

      - The primary network interface address is 10.0.0.102, and its name is eth0.
      - The extended network interface address is 10.0.1.191, and its name is eth1.


```
[root@ecs-resource ~]# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.0.0.102  netmask 255.255.255.0  broadcast 10.0.0.255
        inet6 fe80::f816:3eff:fe22:2288  prefixlen 64  scopeid 0x20<link>
        ether fa:16:3e:22:22:88  txqueuelen 1000  (Ethernet)
        RX packets 135116  bytes 132321802 (126.1 MiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 60963  bytes 23201005 (22.1 MiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

eth1: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.0.1.191  netmask 255.255.255.0  broadcast 10.0.1.255
        inet6 fe80::f816:3eff:fe22:23e1  prefixlen 64  scopeid 0x20<link>
        ether fa:16:3e:22:23:e1  txqueuelen 1000  (Ethernet)
        RX packets 885  bytes 97676 (95.3 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 47  bytes 4478 (4.3 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

c. Configure the **ifcfg** file of the primary network interface.

   i. Run the following command to open the **ifcfg** file of the primary network interface:

      vi /etc/sysconfig/network-scripts/ifcfg-*Primary network interface name*

      The name of the primary network interface is obtained in **3.b**.

      In this example, run the following command:

      vi /etc/sysconfig/network-scripts/ifcfg-eth0

   ii. Press **i** to enter the editing mode.

   iii. Add the following content to the end of the file:

   ```
   IPV6INIT="yes"
   DHCPV6C="yes"
   ```

   iv. Press **ESC** to exit and enter **:wq!** to save the configuration.

d. Configure the **ifcfg** file of the extended network interface.

   i. Run the following command to open the **ifcfg** file of the extended network interface:

      vi /etc/sysconfig/network-scripts/ifcfg-*Extended network interface name*

      The name of the extended network interface is obtained in **3.b**.

      In this example, run the following command:

      vi /etc/sysconfig/network-scripts/ifcfg-eth1

   ii. Press **i** to enter the editing mode.

   iii. Add the following content to the end of the file:

   ```
   IPV6INIT="yes"
   DHCPV6C="yes"
   ```

   iv. Press **ESC** to exit and enter **:wq!** to save the configuration.

e. Edit the **/etc/sysconfig/network** file.

   i. Run the following command to open the **/etc/sysconfig/network** file:

      vi /etc/sysconfig/network

   ii. Press **i** to enter the editing mode.

   iii. Add the following content to the end of the file:

   ```
   NETWORKING_IPV6="yes"
   ```


iv. Press **ESC** to exit and enter **:wq!** to save the configuration.

f. Run the following command to restart the network service for the configuration to take effect:

systemctl restart NetworkManager

g. Run the following command to check whether the ECS has IPv6 addresses:

ip addr

In the following command output, each network interface has an additional **inet6** entry starting with **2407**, followed by an entry starting with **fe80**. This indicates that the ECS has been assigned IPv6 addresses.

```
[root@ecs-resource ~]# ip addr
...
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group
default qlen 1000
    link/ether fa:16:3e:22:22:88 brd ff:ff:ff:ff:ff:ff
    inet 10.0.0.102/24 brd 10.0.0.255 scope global dynamic noprefixroute eth0
       valid_lft 107999994sec preferred_lft 107999994sec
    inet6 2407:c080:1200:1dd8:859c:e5d5:8b3d:a2d9/128 scope global dynamic noprefixroute
       valid_lft 7195sec preferred_lft 7195sec
    inet6 fe80::f816:3eff:fe22:2288/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
3: eth1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group
default qlen 1000
    link/ether fa:16:3e:22:23:e1 brd ff:ff:ff:ff:ff:ff
    inet 10.0.1.191/24 brd 10.0.1.255 scope global dynamic noprefixroute eth1
       valid_lft 107999994sec preferred_lft 107999994sec
    inet6 2407:c080:1200:1a9c:7cc0:63b5:8e65:4dd8/128 scope global dynamic noprefixroute
       valid_lft 7198sec preferred_lft 7198sec
    inet6 fe80::f816:3eff:fe22:23e1/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
```

h. Log in to the destination ECS and assign an IPv6 address by performing operations from **3.a** to **3.g**.

4. Log in to the source ECS and check whether it can use its primary network interface to communicate with the destination ECS:

Before configuring policy-based routes, ensure that the source ECS can use its primary network interface to communicate with the destination ECS.

ping6 -I *<IP-address-of-the-primary-network-interface-on-the-source-ECS> <IP-address-of-the-destination-ECS>*

In this example, run the following command:

ping6 -I 2407:c080:1200:1dd8:859c:e5d5:8b3d:a2d9 2407:c080:1200:1dd9:16a7:fe7a:8f71:7044

If information similar to the following is displayed, the source ECS can use its primary network interface to communicate with the destination ECS.

```
[root@ecs-resource ~]# ping6 -I 2407:c080:1200:1dd8:859c:e5d5:8b3d:a2d9
2407:c080:1200:1dd9:16a7:fe7a:8f71:7044
PING 2407:c080:1200:1dd9:16a7:fe7a:8f71:7044(2407:c080:1200:1dd9:16a7:fe7a:8f71:7044) from
2407:c080:1200:1dd8:859c:e5d5:8b3d:a2d9 : 56 data bytes
64 bytes from 2407:c080:1200:1dd9:16a7:fe7a:8f71:7044: icmp_seq=1 ttl=64 time=0.635 ms
64 bytes from 2407:c080:1200:1dd9:16a7:fe7a:8f71:7044: icmp_seq=2 ttl=64 time=0.320 ms
64 bytes from 2407:c080:1200:1dd9:16a7:fe7a:8f71:7044: icmp_seq=3 ttl=64 time=0.287 ms
64 bytes from 2407:c080:1200:1dd9:16a7:fe7a:8f71:7044: icmp_seq=4 ttl=64 time=0.193 ms
^C
--- 2407:c080:1200:1dd9:16a7:fe7a:8f71:7044 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3074ms
rtt min/avg/max/mdev = 0.193/0.358/0.635/0.167 ms
```


5. Log in to the source ECS and configure temporary routes for the ECS.

NOTICE

Temporary routes are applied immediately but are lost after ECS restarts. To avoid network disruptions, perform **6** to configure permanent routes instead.

a. Configure policy-based routes for both the primary and extended network interfaces.

- Primary network interface

  ip -6 route add default via *<subnet-gateway>* **dev** *<network-interface-name>* **table** *<route-table-name>*

  ip -6 route add *<subnet-CIDR-block>* **dev** *<network-interface-name>* **table** *<route-table-name>*

  ip -6 rule add from *<network-interface-address>* **table** *<route-table-name>*

- Extended network interface

  ip -6 route add default via *<subnet-gateway>* **dev** *<network-interface-name>* **table** *<route-table-name>*

  ip -6 route add *<subnet-CIDR-block>* **dev** *<network-interface-name>* **table** *<route-table-name>*

  ip -6 rule add from *<network-interface-address>* **table** *<route-table-name>*

Configure the parameters as follows:

- Network interface name: Enter the name obtained in **3.b**.
- Route table name: Name the route table with a number.
- Other network information: Enter the IP addresses collected in **1**.

In this example, run the following commands:

- Primary network interface

  ip -6 route add default via 2407:c080:1200:1dd8::1 dev eth0 table 10

  ip -6 route add 2407:c080:1200:1dd8::/64 dev eth0 table 10

  ip -6 rule add from 2407:c080:1200:1dd8:859c:e5d5:8b3d:a2d9 table 10

- Extended network interface

  ip -6 route add default via 2407:c080:1200:1a9c::1 dev eth1 table 20

  ip -6 route add 2407:c080:1200:1a9c::/64 dev eth1 table 20

  ip -6 rule add from 2407:c080:1200:1a9c:7cc0:63b5:8e65:4dd8 table 20

NOTE

If the ECS has multiple network interfaces, configure policy-based routes for all network interfaces one by one.

b. Check whether the policy-based routes are added.

ip -6 rule

ip -6 route show table *<route-table-name-of-the-primary-network-interface>*

ip -6 route show table *<route-table-name-of-the-extended-network-interface>*

The route table name is the one configured in **5.a**.

In this example, run the following commands:

ip -6 rule

ip -6 route show table 10

ip -6 route show table 20

If information similar to the following is displayed, the policy-based routes have been added.

```
[root@ecs-resource ~]# ip -6 rule
0:      from all lookup local
32764:  from 2407:c080:1200:1a9c:7cc0:63b5:8e65:4dd8 lookup 20
32765:  from 2407:c080:1200:1dd8:859c:e5d5:8b3d:a2d9 lookup 10
32766:  from all lookup main
[root@ecs-resource ~]# ip -6 route show table 10
2407:c080:1200:1dd8::/64 dev eth0 metric 1024 pref medium
default via 2407:c080:1200:1dd8::1 dev eth0 metric 1024 pref medium
[root@ecs-resource ~]# ip -6 route show table 20
2407:c080:1200:1a9c::/64 dev eth1 metric 1024 pref medium
default via 2407:c080:1200:1a9c::1 dev eth1 metric 1024 pref medium
```

c. Check whether the source and destination ECSs can communicate with each other.

ping -6 -I *<IP-address-of-the-primary-network-interface-on-the-source-ECS> <IP-address-of-the-destination-ECS>*

ping -6 -I *<IP-address-of-the-extended-network-interface-on-the-source-ECS> <IP-address-of-the-destination-ECS>*

In this example, run the following commands:

ping -6 -I 2407:c080:1200:1dd8:859c:e5d5:8b3d:a2d9 2407:c080:1200:1dd9:16a7:fe7a:8f71:7044

ping -6 -I 2407:c080:1200:1a9c:7cc0:63b5:8e65:4dd8 2407:c080:1200:1dd9:16a7:fe7a:8f71:7044

If information similar to the following is displayed, both the network interfaces of the source ECS can communicate with the destination ECS.

```
[root@ecs-resource ~]# ping -6 -I 2407:c080:1200:1dd8:859c:e5d5:8b3d:a2d9
2407:c080:1200:1dd9:16a7:fe7a:8f71:7044
PING 2407:c080:1200:1dd9:16a7:fe7a:8f71:7044(2407:c080:1200:1dd9:16a7:fe7a:8f71:7044) from
2407:c080:1200:1dd8:859c:e5d5:8b3d:a2d9 : 56 data bytes
64 bytes from 2407:c080:1200:1dd9:16a7:fe7a:8f71:7044: icmp_seq=1 ttl=64 time=0.770 ms
64 bytes from 2407:c080:1200:1dd9:16a7:fe7a:8f71:7044: icmp_seq=2 ttl=64 time=0.295 ms
64 bytes from 2407:c080:1200:1dd9:16a7:fe7a:8f71:7044: icmp_seq=3 ttl=64 time=0.245 ms
^C
--- 2407:c080:1200:1dd9:16a7:fe7a:8f71:7044 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2080ms
rtt min/avg/max/mdev = 0.245/0.436/0.770/0.237 ms
[root@ecs-resource ~]# ping -6 -I 2407:c080:1200:1a9c:7cc0:63b5:8e65:4dd8
2407:c080:1200:1dd9:16a7:fe7a:8f71:7044
```



```
PING 2407:c080:1200:1dd9:16a7:fe7a:8f71:7044(2407:c080:1200:1dd9:16a7:fe7a:8f71:7044) from 2407:c080:1200:1a9c:7cc0:63b5:8e65:4dd8 : 56 data bytes
64 bytes from 2407:c080:1200:1dd9:16a7:fe7a:8f71:7044: icmp_seq=1 ttl=64 time=0.922 ms
64 bytes from 2407:c080:1200:1dd9:16a7:fe7a:8f71:7044: icmp_seq=2 ttl=64 time=0.307 ms
64 bytes from 2407:c080:1200:1dd9:16a7:fe7a:8f71:7044: icmp_seq=3 ttl=64 time=0.174 ms
^C
--- 2407:c080:1200:1dd9:16a7:fe7a:8f71:7044 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2059ms
rtt min/avg/max/mdev = 0.174/0.467/0.922/0.326 ms
```

6. Configure permanent routes for the source ECS.

   a. Run the following command to open the **/etc/rc.local** file:

      vi /etc/rc.local

   b. Press **i** to enter the editing mode.

   c. Add the following content to the end of the file:

```
# check eth0
for ((x=0; x<10; x++)); do
  if (ip addr show eth0 | grep -w 2407:c080:1200:1dd8:859c:e5d5:8b3d:a2d9 >/dev/null 2>&1);
then
    break
  fi
  sleep 1
done

# Add v6 routes for eth0
ip -6 route flush table 10
ip -6 route add default via 2407:c080:1200:1dd8::1 dev eth0 table 10
ip -6 route add 2407:c080:1200:1dd8::/64 dev eth0 table 10
ip -6 rule add from 2407:c080:1200:1dd8:859c:e5d5:8b3d:a2d9 table 10

# check eth1
for ((x=0; x<10; x++)); do
  if (ip addr show eth1 | grep -w 2407:c080:1200:1a9c:7cc0:63b5:8e65:4dd8 >/dev/null 2>&1);
then
    break
  fi
  sleep 1
done

# Add v6 routes for eth1
ip -6 route flush table 20
ip -6 route add default via 2407:c080:1200:1a9c::1 dev eth1 table 20
ip -6 route add 2407:c080:1200:1a9c::/64 dev eth1 table 20
ip -6 rule add from 2407:c080:1200:1a9c:7cc0:63b5:8e65:4dd8 table 20
```

   The parameters are as follows:

   - **check eth0**: Check whether primary network interface eth0 has obtained an IPv6 address (2407:c080:1200:1dd8:859c:e5d5:8b3d:a2d9) every second up to 10 times.
   - **Add v6 routes for eth0**: Add policy-based routes for the primary network interface. Set the value to be the same as that configured in **5.a**.
   - **check eth1**: Check whether extended network interface eth1 has obtained an IPv6 address (2407:c080:1200:1a9c:7cc0:63b5:8e65:4dd8) every second up to 10 times.
   - **Add v6 routes for eth1**: Add policy-based routes for the extended network interface. Set the value to be the same as that configured in **5.a**.

d. Press **ESC** to exit and enter **:wq!** to save the configuration.

e. Run the following command to assign execute permissions to the **/etc/rc.local** file:

chmod +x /etc/rc.local

NOTE

If your operating system is Red Hat or EulerOS, run the following command after you perform **6.e**:

chmod +x /etc/rc.d/rc.local

f. Run the following command to restart the ECS:

reboot

NOTICE

Policy-based routes added to the **/etc/rc.local** file take effect only after the ECS is restarted. Ensure that workloads on the ECS will not be affected before restarting the ECS.

g. Repeat **5.b** to **5.c** to check whether the policy-based routes are added and whether the source ECS and the destination ECS can communicate with each other.

5.3.2.4 Configuring IPv4 and IPv6 Policy-based Routes for a Linux ECS with Multiple Network Interfaces (Ubuntu)

Scenarios

This section describes how to configure policy-based routes for an Ubuntu 22.04 server 64-bit ECS with two network interfaces.

- IPv4: **Configuring IPv4 Policy-based Routes for an Ubuntu ECS**
- IPv6: **Configuring IPv6 Policy-based Routes for an Ubuntu ECS**

For details about the background knowledge and networking of an ECS with two network interfaces, see **Overview**.

Configuring IPv4 Policy-based Routes for an Ubuntu ECS

1. Collect the ECS network information required for configuring policy-based routes.

   For details, see **Collecting ECS Network Information**.

   In this example, the network information of the ECS is shown in **Table 5-15**.

Table 5-15 Ubuntu ECS using IPv4

ECS	Primary Network Interface	Extended Network Interface
Source	• IP address: 10.0.0.138 • Subnet: 10.0.0.0/24 • Subnet gateway: 10.0.0.1	• IP address: 10.0.1.25 • Subnet: 10.0.1.0/24 • Subnet gateway: 10.0.1.1
Destination	IP address: 10.0.2.146	N/A

2. Log in to the source ECS.

   For details, see **How Do I Log In to My ECS?**

3. Check whether the source ECS can use its primary network interface to communicate with the destination ECS:

   Before configuring policy-based routes, ensure that the source ECS can use its primary network interface to communicate with the destination ECS.

   ping -I *<IP-address-of-the-primary-network-interface-on-the-source-ECS> <IP-address-of-the-destination-ECS>*

   In this example, run the following command:

   ping -I 10.0.0.138 10.0.2.146

   If information similar to the following is displayed, the source ECS can use its primary network interface to communicate with the destination ECS.

```
root@ecs-s:~# ping -I 10.0.0.138 10.0.2.146
PING 10.0.2.146 (10.0.2.146) from 10.0.0.138 : 56(84) bytes of data.
64 bytes from 10.0.2.146: icmp_seq=1 ttl=64 time=0.247 ms
64 bytes from 10.0.2.146: icmp_seq=2 ttl=64 time=0.194 ms
64 bytes from 10.0.2.146: icmp_seq=3 ttl=64 time=0.190 ms
^C
--- 10.0.2.146 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2049ms
rtt min/avg/max/mdev = 0.190/0.210/0.247/0.025 ms
```

4. Query the network interface names of the source ECS:

   ip addr

   Search for the network interface names based on IP addresses.

   – The primary network interface address is 10.0.0.138, and its name is eth0.

   – The extended network interface address is 10.0.1.25, and its name is eth1.

```
root@ecs-s:~# ip addr
...
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP group default qlen 1000
    link/ether fa:16:3e:22:22:ac brd ff:ff:ff:ff:ff:ff
    altname enp0s3
    altname ens3
    inet 10.0.0.138/24 brd 10.0.0.255 scope global dynamic noprefixroute eth0
       valid_lft 107999167sec preferred_lft 107999167sec
    inet6 fe80::f816:3eff:fe22:22ac/64 scope link
       valid_lft forever preferred_lft forever
3: eth1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP group default qlen 1000
    link/ether fa:16:3e:22:23:3b brd ff:ff:ff:ff:ff:ff
    altname enp4s1
```



```
inet 10.0.1.25/24 brd 10.0.1.255 scope global dynamic noprefixroute eth1
   valid_lft 107999167sec preferred_lft 107999167sec
inet6 fe80::f816:3eff:fe22:233b/64 scope link
   valid_lft forever preferred_lft forever
```

5. Configure temporary routes for the source ECS.

NOTICE

Temporary routes are applied immediately but are lost after ECS restarts. To avoid network disruptions, perform **6** to configure permanent routes instead.

a. Configure policy-based routes for both the primary and extended network interfaces.

- Primary network interface

  ip route add default via *<subnet-gateway>* **dev** *<network-interface-name>* **table** *<route-table-name>*

  ip route add *<subnet-CIDR-block>* **dev** *<network-interface-name>* **table** *<route-table-name>*

  ip rule add from *<network-interface-address>* **table** *<route-table-name>*

- Extended network interface

  ip route add default via *<subnet-gateway>* **dev** *<network-interface-name>* **table** *<route-table-name>*

  ip route add *<subnet-CIDR-block>* **dev** *<network-interface-name>* **table** *<route-table-name>*

  ip rule add from *<network-interface-address>* **table** *<route-table-name>*

Configure the parameters as follows:

- Network interface name: Enter the name obtained in **4**.
- Route table name: Name the route table with a number.
- Other network information: Enter the IP addresses collected in **1**.

In this example, run the following commands:

- Primary network interface

  ip route add default via 10.0.0.1 dev eth0 table 10

  ip route add 10.0.0.0/24 dev eth0 table 10

  ip rule add from 10.0.0.138 table 10

- Extended network interface

  ip route add default via 10.0.1.1 dev eth1 table 20

  ip route add 10.0.1.0/24 dev eth1 table 20

  ip rule add from 10.0.1.25 table 20

NOTE

If the ECS has multiple network interfaces, configure policy-based routes for all network interfaces one by one.

b. Check whether the policy-based routes are added.

ip rule

ip route show table *<route-table-name-of-the-primary-network-interface>*

ip route show table *<route-table-name-of-the-extended-network-interface>*

The route table name is the one configured in **5.a**.

In this example, run the following commands:

ip rule

ip route show table 10

ip route show table 20

If information similar to the following is displayed, the policy-based routes have been added.

```
root@ecs-s:~# ip rule
0:      from all lookup local
32764:  from 10.0.1.25 lookup 20
32765:  from 10.0.0.138 lookup 10
32766:  from all lookup main
32767:  from all lookup default
root@ecs-s:~# ip route show table 10
default via 10.0.0.1 dev eth0
10.0.0.0/24 dev eth0 scope link
root@ecs-s:~# ip route show table 20
default via 10.0.1.1 dev eth1
10.0.1.0/24 dev eth1 scope link
```

c. Check whether the source and destination ECSs can communicate with each other.

ping -I *<IP-address-of-the-primary-network-interface-on-the-source-ECS> <IP-address-of-the-destination-ECS>*

ping -I *<IP-address-of-the-extended-network-interface-on-the-source-ECS> <IP-address-of-the-destination-ECS>*

In this example, run the following commands:

ping -I 10.0.0.138 10.0.2.146

ping -I 10.0.1.25 10.0.2.146

If information similar to the following is displayed, both the network interfaces of the source ECS can communicate with the destination ECS.

```
root@ecs-s:~# ping -I 10.0.0.138 10.0.2.146
PING 10.0.2.146 (10.0.2.146) from 10.0.0.138 : 56(84) bytes of data.
64 bytes from 10.0.2.146: icmp_seq=1 ttl=64 time=0.258 ms
64 bytes from 10.0.2.146: icmp_seq=2 ttl=64 time=0.242 ms
64 bytes from 10.0.2.146: icmp_seq=3 ttl=64 time=0.165 ms
^C
--- 10.0.2.146 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2039ms
rtt min/avg/max/mdev = 0.165/0.221/0.258/0.040 ms
root@ecs-s:~# ping -I 10.0.1.25 10.0.2.146
PING 10.0.2.146 (10.0.2.146) from 10.0.1.25 : 56(84) bytes of data.
64 bytes from 10.0.2.146: icmp_seq=1 ttl=64 time=0.498 ms
64 bytes from 10.0.2.146: icmp_seq=2 ttl=64 time=0.427 ms
64 bytes from 10.0.2.146: icmp_seq=3 ttl=64 time=0.185 ms
```



```
^C
--- 10.0.2.146 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2031ms
rtt min/avg/max/mdev = 0.185/0.370/0.498/0.133 ms
```

6. Configure permanent routes for the source ECS.

   a. Run the following command to add **network-routes.service** to the systemd service:

      vi /etc/systemd/system/network-routes.service

   b. Press **i** to enter the editing mode.

   c. Add the following content to the end of the file:

```
[Unit]
Description=Network Routes Configuration
After=network.target

[Service]
Type=oneshot
RemainAfterExit=yes
ExecStart=/bin/bash -c 'for((x=0; x<10; x++)); do [[ $(ip addr show eth0 | grep -w 10.0.0.138 >/dev/null 2>&1 && echo 1) ]] && break; sleep 1; done; ip route flush table 10; ip route add default via  10.0.0.1 dev eth0 table 10; ip route add 10.0.0.0/24 dev eth0 table 10; ip rule add from 10.0.0.138 table 10; for((x=0; x<10; x++)); do [[ $(ip addr show eth1 | grep -w 10.0.1.25 >/dev/null 2>&1 && echo 1) ]] && break; sleep 1; done; ip route flush table 20; ip route add default via 10.0.1.1 dev eth1 table 20; ip route add 10.0.1.0/24 dev eth1 table 20; ip rule add from 10.0.1.25 table 20; ip rule add to 169.254.169.254 table main'

[Install]
WantedBy=multi-user.target
```

      The parameters are as follows:

      - for loop: Check whether eth0 or eth1 has obtained an IPv4 address (eth0: 10.0.0.138; eth1: 10.0.1.25) every second up to 10 times.
      - **ip route flush table** *route-table-name*: Running this command will delete existing routes in the specified route table. This prevents new routes from being affected.
      - Policy-based routes of the primary network interface: Set it to the same value as that in **5.a**.
      - Policy-based routes of the extended network interface: Set it to the same value as that in **5.a**.
      - **ip rule add to 169.254.169.254 table main**: Set the Cloud-Init address to the same value as that in this example.

   d. Press **ESC** to exit and enter **:wq!** to save the configuration.

   e. Run the following commands to reload the systemd configuration and start the service:

      systemctl daemon-reload

      systemctl enable network-routes.service

      If information similar to the following is displayed, the service is started:

```
root@ecs-s:~# systemctl daemon-reload
root@ecs-s:~# systemctl enable network-routes.service
Created symlink /etc/systemd/system/multi-user.target.wants/network-routes.service → /etc/systemd/system/network-routes.service.
```

   f. Run the following command to restart the source ECS:

      reboot

NOTICE

Policy-based routes added to the **network-routes.service** file only work after the source ECS is restarted. Ensure that workloads on the ECS will not be affected before restarting the ECS.

g. Repeat **5.b** to **5.c** to check whether the policy-based routes are added and whether the source ECS and the destination ECS can communicate with each other.

Configuring IPv6 Policy-based Routes for an Ubuntu ECS

1. Collect the ECS network information required for configuring policy-based routes.

   For details, see **Collecting ECS Network Information**.

   In this example, the network information of the ECS is shown in **Table 5-16**.

   Table 5-16 Ubuntu ECS using IPv6

ECS	Primary Network Interface	Extended Network Interface
Source	• IPv4 address: 10.0.0.138 • IPv6 address: 2407:c080:1200:1dd8:1473:49db:22d7:13c7 • IPv6 subnet: 2407:c080:1200:1dd8::/64 • IPv6 subnet gateway: 2407:c080:1200:1dd8::1	• IPv4 address: 10.0.1.25 • IPv6 address: 2407:c080:1200:1a9c:691e:fffe:7e22:12c4 • IPv6 subnet: 2407:c080:1200:1a9c::/64 • IPv6 subnet gateway: 2407:c080:1200:1a9c::1
Destination	• IPv4 address: 10.0.2.146 • IPv6 address: 2407:c080:1200:1dd9:f5e1:94d1:2822:dede	N/A

2. Log in to the source ECS.

   For details, see **How Do I Log In to My ECS?**

3. Check whether the ECSs have IPv6 enabled and have IPv6 addresses.

NOTICE

Perform this step for both the source and destination ECSs to ensure that the ECSs have IPv6 addresses. Otherwise, the ECSs cannot communicate with each other using IPv6 addresses.

ECSs in this example run Ubuntu 22.04 server (64-bit). For details about how to assign IPv6 addresses for ECSs running other OSs, see **Dynamically Assigning IPv6 Addresses**.

a. Run the following command to check whether the source ECS has IPv6 addresses:

ip addr

In the following command output, eth0 and eth1 are the network interfaces of the ECS. Each network interface has one **inet6** entry starting with **fe80**. This indicates that the ECS has IPv6 enabled but has not been assigned IPv6 addresses. In this case, perform **3.b** to **3.h** to assign IPv6 addresses.

```
root@ecs-s:~# ip addr
...
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP group default
qlen 1000
    link/ether fa:16:3e:22:22:ac brd ff:ff:ff:ff:ff:ff
    altname enp0s3
    altname ens3
    inet 10.0.0.138/24 brd 10.0.0.255 scope global dynamic noprefixroute eth0
       valid_lft 107999781sec preferred_lft 107999781sec
    inet6 fe80::f816:3eff:fe22:22ac/64 scope link
       valid_lft forever preferred_lft forever
3: eth1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP group default
qlen 1000
    link/ether fa:16:3e:22:23:3b brd ff:ff:ff:ff:ff:ff
    altname enp4s1
    inet 10.0.1.25/24 brd 10.0.1.255 scope global dynamic noprefixroute eth1
       valid_lft 107999781sec preferred_lft 107999781sec
    inet6 fe80::f816:3eff:fe22:233b/64 scope link
       valid_lft forever preferred_lft forever
```

b. Query the network interface names of the source ECS:

ifconfig

Search for the network interface names based on IP addresses.

- The primary network interface address is 10.0.0.138, and its name is eth0.
- The extended network interface address is 10.0.1.25, and its name is eth1.

```
root@ecs-s:~# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.0.0.138  netmask 255.255.255.0  broadcast 10.0.0.255
        inet6 fe80::f816:3eff:fe22:22ac  prefixlen 64  scopeid 0x20<link>
        ether fa:16:3e:22:22:ac  txqueuelen 1000  (Ethernet)
        RX packets 863  bytes 269089 (269.0 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 1117  bytes 359807 (359.8 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

eth1: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.0.1.25  netmask 255.255.255.0  broadcast 10.0.1.255
        inet6 fe80::f816:3eff:fe22:233b  prefixlen 64  scopeid 0x20<link>
        ether fa:16:3e:22:23:3b  txqueuelen 1000  (Ethernet)
        RX packets 10  bytes 1358 (1.3 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 10  bytes 973 (973.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
...
```

c. Configure the **01-netcfg.yaml** file.

i. Run the following command to access **/etc/netplan/**:

cd /etc/netplan

ii. Run the following command to open the **01-netcfg.yaml** file:

vi 01-netcfg.yaml

iii. Press **i** to enter the editing mode.

iv. Add **dhcp6: true** to the network interfaces for which you want to assign IPv6 addresses as follows:

In this example, the primary network interface name queried in **3.b** is eth0, and the extended network interface name is eth1.

```
network:
  version: 2
  renderer: NetworkManager
  ethernets:
    eth0:
      dhcp4: true
      dhcp6: true
    eth1:
      dhcp4: true
      dhcp6: true
    eth2:
      dhcp4: true
    eth3:
      dhcp4: true
    eth4:
      dhcp4: true
```

v. Press **ESC** to exit and enter **:wq!** to save the configuration.

d. Run the following commands to change the permissions on the **01-netcfg.yaml** file and ensure that only the file owner has the read and write permissions:

chmod 600 /etc/netplan/01-netcfg.yaml

chown root:root /etc/netplan/01-netcfg.yaml

e. Run the following command to apply the modification:

netplan apply

f. Configure the **NetworkManager.conf** file.

i. Run the following command to open the **NetworkManager.conf** file:

vi /etc/NetworkManager/NetworkManager.conf

ii. Press **i** to enter the editing mode.

iii. Add **dhcp=dhclient** to the file as follows:

```
[main]
plugins=ifupdown,keyfile
dhcp=dhclient

[ifupdown]
managed=true

[device]
wifi.scan-rand-mac-address=no
```

iv. Press **ESC** to exit and enter **:wq!** to save the configuration.

g. Run the following command to restart the network service for the configuration take effect:

systemctl restart NetworkManager

h. Run the following command to check whether the source ECS has IPv6 addresses:

ip addr

In the following command output, each network interface has an additional **inet6** entry starting with **2407**, followed by an entry starting with **fe80**. This indicates that the ECS has been assigned IPv6 addresses.

```
root@ecs-s:/etc/netplan# ip addr
...
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP group default
qlen 1000
    link/ether fa:16:3e:22:22:ac brd ff:ff:ff:ff:ff:ff
    altname enp0s3
    altname ens3
    inet 10.0.0.138/24 brd 10.0.0.255 scope global dynamic noprefixroute eth0
       valid_lft 107999982sec preferred_lft 107999982sec
    inet6 2407:c080:1200:1dd8:1473:49db:22d7:13c7/128 scope global dynamic noprefixroute
       valid_lft 7182sec preferred_lft 7182sec
    inet6 fe80::f816:3eff:fe22:22ac/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
3: eth1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP group default
qlen 1000
    link/ether fa:16:3e:22:23:3b brd ff:ff:ff:ff:ff:ff
    altname enp4s1
    inet 10.0.1.25/24 brd 10.0.1.255 scope global dynamic noprefixroute eth1
       valid_lft 107999982sec preferred_lft 107999982sec
    inet6 2407:c080:1200:1a9c:691e:fffe:7e22:12c4/128 scope global dynamic noprefixroute
       valid_lft 7182sec preferred_lft 7182sec
    inet6 fe80::f816:3eff:fe22:233b/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
```

i. Log in to the destination ECS and assign an IPv6 address by performing operations from **3.a** to **3.h**.

In the following command output, **eth0** has an additional **inet6** entry starting with **2407**, followed by an entry starting with **fe80**. This indicates that the ECS has been assigned an IPv6 address.

```
root@ecs-d:/etc/netplan# ip addr
...
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP group default
qlen 1000
    link/ether fa:16:3e:22:24:b4 brd ff:ff:ff:ff:ff:ff
    altname enp0s3
    altname ens3
    inet 10.0.2.146/24 brd 10.0.2.255 scope global dynamic noprefixroute eth0
       valid_lft 107999994sec preferred_lft 107999994sec
    inet6 2407:c080:1200:1dd9:f5e1:94d1:2822:dede/128 scope global dynamic noprefixroute
       valid_lft 7195sec preferred_lft 7195sec
    inet6 fe80::f816:3eff:fe22:24b4/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
```

4. Log in to the source ECS and check whether it can use its primary network interface to communicate with the destination ECS:

Before configuring policy-based routes, ensure that the source ECS can use its primary network interface to communicate with the destination ECS.

ping6 -I *<IP-address-of-the-primary-network-interface-on-the-source-ECS> <IP-address-of-the-destination-ECS>*

In this example, run the following command:

ping6 -I 2407:c080:1200:1dd8:1473:49db:22d7:13c7 2407:c080:1200:1dd9:f5e1:94d1:2822:dede

If information similar to the following is displayed, the source ECS can use its primary network interface to communicate with the destination ECS.

```
root@ecs-s:/etc/netplan# ping6 -I 2407:c080:1200:1dd8:1473:49db:22d7:13c7
2407:c080:1200:1dd9:f5e1:94d1:2822:dede
PING 2407:c080:1200:1dd9:f5e1:94d1:2822:dede(2407:c080:1200:1dd9:f5e1:94d1:2822:dede) from
2407:c080:1200:1dd8:1473:49db:22d7:13c7 : 56 data bytes
```



```
64 bytes from 2407:c080:1200:1dd9:f5e1:94d1:2822:dede: icmp_seq=1 ttl=64 time=0.244 ms
64 bytes from 2407:c080:1200:1dd9:f5e1:94d1:2822:dede: icmp_seq=2 ttl=64 time=0.212 ms
64 bytes from 2407:c080:1200:1dd9:f5e1:94d1:2822:dede: icmp_seq=3 ttl=64 time=0.169 ms
^C
--- 2407:c080:1200:1dd9:f5e1:94d1:2822:dede ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2033ms
rtt min/avg/max/mdev = 0.169/0.208/0.244/0.030 ms
```

5. Log in to the source ECS and configure temporary routes for the ECS.

NOTICE

Temporary routes are applied immediately but are lost after ECS restarts. To avoid network disruptions, perform **6** to configure permanent routes instead.

a. Configure policy-based routes for both the primary and extended network interfaces.

- Primary network interface

  ip -6 route add default via *<subnet-gateway>* **dev** *<network-interface-name>* **table** *<route-table-name>*

  ip -6 route add *<subnet-CIDR-block>* **dev** *<network-interface-name>* **table** *<route-table-name>*

  ip -6 rule add from *<network-interface-address>* **table** *<route-table-name>*

- Extended network interface

  ip -6 route add default via *<subnet-gateway>* **dev** *<network-interface-name>* **table** *<route-table-name>*

  ip -6 route add *<subnet-CIDR-block>* **dev** *<network-interface-name>* **table** *<route-table-name>*

  ip -6 rule add from *<network-interface-address>* **table** *<route-table-name>*

Configure the parameters as follows:

- Network interface name: Enter the name obtained in **3.b**.
- Route table name: Name the route table with a number.
- Other network information: Enter the IP addresses collected in **1**.

In this example, run the following commands:

- Primary network interface

  ip -6 route add default via 2407:c080:1200:1dd8::1 dev eth0 table 10

  ip -6 route add 2407:c080:1200:1dd8::/64 dev eth0 table 10

  ip -6 rule add from 2407:c080:1200:1dd8:1473:49db:22d7:13c7 table 10

- Extended network interface

  ip -6 route add default via 2407:c080:1200:1a9c::1 dev eth1 table 20

ip -6 route add 2407:c080:1200:1a9c::/64 dev eth1 table 20

ip -6 rule add from 2407:c080:1200:1a9c:691e:fffe:7e22:12c4 table 20

NOTE

If the ECS has multiple network interfaces, configure policy-based routes for all network interfaces one by one.

b. Check whether the policy-based routes are added.

ip -6 rule

ip -6 route show table *<route-table-name-of-the-primary-network-interface>*

ip -6 route show table *<route-table-name-of-the-extended-network-interface>*

The route table name is the one configured in **5.a**.

In this example, run the following commands:

ip -6 rule

ip -6 route show table 10

ip -6 route show table 20

If information similar to the following is displayed, the policy-based routes have been added.

```
root@ecs-s:/etc/netplan# ip -6 rule
0:      from all lookup local
32764:  from 2407:c080:1200:1a9c:691e:fffe:7e22:12c4 lookup 20
32765:  from 2407:c080:1200:1dd8:1473:49db:22d7:13c7 lookup 10
32766:  from all lookup main
root@ecs-s:/etc/netplan# ip -6 route show table 10
2407:c080:1200:1dd8::/64 dev eth0 metric 1024 pref medium
default via 2407:c080:1200:1dd8::1 dev eth0 metric 1024 pref medium
root@ecs-s:/etc/netplan# ip -6 route show table 20
2407:c080:1200:1a9c::/64 dev eth1 metric 1024 pref medium
default via 2407:c080:1200:1a9c::1 dev eth1 metric 1024 pref medium
```

c. Check whether the source and destination ECSs can communicate with each other.

ping -6 -I *<IP-address-of-the-primary-network-interface-on-the-source-ECS> <IP-address-of-the-destination-ECS>*

ping -6 -I *<IP-address-of-the-extended-network-interface-on-the-source-ECS> <IP-address-of-the-destination-ECS>*

In this example, run the following commands:

ping6 -I 2407:c080:1200:1dd8:1473:49db:22d7:13c7 2407:c080:1200:1dd9:f5e1:94d1:2822:dede

ping6 -I 2407:c080:1200:1a9c:691e:fffe:7e22:12c4 2407:c080:1200:1dd9:f5e1:94d1:2822:dede

If information similar to the following is displayed, both the network interfaces of the source ECS can communicate with the destination ECS.

```
root@ecs-s:/etc/netplan# ping6 -I 2407:c080:1200:1dd8:1473:49db:22d7:13c7
2407:c080:1200:1dd9:f5e1:94d1:2822:dede
PING 2407:c080:1200:1dd9:f5e1:94d1:2822:dede(2407:c080:1200:1dd9:f5e1:94d1:2822:dede)
from 2407:c080:1200:1dd8:1473:49db:22d7:13c7 : 56 data bytes
64 bytes from 2407:c080:1200:1dd9:f5e1:94d1:2822:dede: icmp_seq=1 ttl=64 time=0.260 ms
64 bytes from 2407:c080:1200:1dd9:f5e1:94d1:2822:dede: icmp_seq=2 ttl=64 time=0.248 ms
64 bytes from 2407:c080:1200:1dd9:f5e1:94d1:2822:dede: icmp_seq=3 ttl=64 time=0.165 ms
^C
```



```
--- 2407:c080:1200:1dd9:f5e1:94d1:2822:dede ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2043ms
rtt min/avg/max/mdev = 0.165/0.224/0.260/0.042 ms
root@ecs-s:/etc/netplan# ping6 -I 2407:c080:1200:1a9c:691e:fffe:7e22:12c4
2407:c080:1200:1dd9:f5e1:94d1:2822:dede
PING 2407:c080:1200:1dd9:f5e1:94d1:2822:dede(2407:c080:1200:1dd9:f5e1:94d1:2822:dede)
from 2407:c080:1200:1a9c:691e:fffe:7e22:12c4 : 56 data bytes
64 bytes from 2407:c080:1200:1dd9:f5e1:94d1:2822:dede: icmp_seq=1 ttl=64 time=0.592 ms
64 bytes from 2407:c080:1200:1dd9:f5e1:94d1:2822:dede: icmp_seq=2 ttl=64 time=0.208 ms
64 bytes from 2407:c080:1200:1dd9:f5e1:94d1:2822:dede: icmp_seq=3 ttl=64 time=0.162 ms
^C
--- 2407:c080:1200:1dd9:f5e1:94d1:2822:dede ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2031ms
rtt min/avg/max/mdev = 0.162/0.320/0.592/0.192 ms
```

6. Configure permanent routes for the source ECS.
   a. Run the following command to create the **network-routes6.service** file for the systemd service:

      vi /etc/systemd/system/network-routes6.service
   b. Press **i** to enter the editing mode.
   c. Add the following content to the end of the file:

```
[Unit]
Description=Network Routes Configuration
After=network.target

[Service]
Type=oneshot
RemainAfterExit=yes

ExecStart=/bin/bash -c 'for((x=0; x<10; x++)); do [[ $(ip addr show eth0 | grep -w
2407:c080:1200:1dd8:1473:49db:22d7:13c7 >/dev/null 2>&1 && echo 1) ]] && break; sleep 1;
done; ip route flush table 10; ip -6 route add default via 2407:c080:1200:1dd8::1 dev eth0 table
10; ip -6 route add 2407:c080:1200:1dd8::/64 dev eth0 table 10; ip -6 rule add from
2407:c080:1200:1dd8:1473:49db:22d7:13c7 table 10; for((x=0; x<10; x++)); do [[ $(ip addr show
eth1 | grep -w 2407:c080:1200:1a9c:691e:fffe:7e22:12c4 >/dev/null 2>&1 && echo 1) ]] &&
break; sleep 1; done; ip route flush table 20; ip -6 route add default via 2407:c080:1200:1a9c::1
dev eth1 table 20; ip -6 route add 2407:c080:1200:1a9c::/64 dev eth1 table 20; ip -6 rule add
from 2407:c080:1200:1a9c:691e:fffe:7e22:12c4 table 20'

[Install]
WantedBy=multi-user.target
```

      The parameters are as follows:

      - for loop: Check whether eth0 or eth1 has obtained an IPv6 address (eth0: 2407:c080:1200:1dd8:1473:49db:22d7:13c7; eth1:2407:c080:1200:1a9c:691e:fffe:7e22:12c4) every second up to 10 times.
      - **ip route flush table** *route-table-name*: Running this command will delete existing routes in the specified route table. This prevents new routes from being affected.
      - Policy-based routes of the primary network interface: Set it to the same value as that in **5.a**.
      - Policy-based routes of the extended network interface: Set it to the same value as that in **5.a**.
   d. Press **ESC** to exit and enter **:wq!** to save the configuration.
   e. Run the following commands to reload the systemd configuration and start the service:

      systemctl daemon-reload

systemctl enable network-routes6.service

If information similar to the following is displayed, the service is started:

```
root@ecs-s:/etc/netplan# systemctl daemon-reload
root@ecs-s:/etc/netplan# systemctl enable network-routes6.service
Created symlink /etc/systemd/system/multi-user.target.wants/network-routes6.service → /etc/systemd/system/network-routes6.service.
```

f. Run the following command to restart the source ECS:

reboot

NOTICE

Policy-based routes added to the **network-routes6.service** file only work after the source ECS is restarted. Ensure that workloads on the ECS will not be affected before restarting the ECS.

g. Repeat 5.b to 5.c to check whether the policy-based routes are added and whether the source ECS and the destination ECS can communicate with each other.

5.3.2.5 Configuring IPv4 and IPv6 Policy-based Routes for a Windows ECS with Multiple Network Interfaces

Scenarios

This section describes how to configure policy-based routes for a Windows Server 2012 64-bit ECS with two network interfaces.

- IPv4: **Configuring IPv4 Policy-based Routes for a Windows ECS**
- IPv6: **Configuring IPv6 Policy-based Routes for a Windows ECS**

For details about the background knowledge and networking of an ECS with two network interfaces, see **Overview**.

Configuring IPv4 Policy-based Routes for a Windows ECS

1. Collect the ECS network information required for configuring policy-based routes.

   For details, see **Collecting ECS Network Information**.

   In this example, the network information of the ECS is shown in **Table 5-17**.

   Table 5-17 Windows ECS using IPv4

ECS	Primary Network Interface	Extended Network Interface
Source	• IP address: 10.0.0.59 • Subnet gateway: 10.0.0.1	• IP address: 10.0.1.104 • Subnet gateway: 10.0.1.1
Destination	IP address: 10.0.2.12	N/A

2. Log in to the source ECS.

   For details, see **How Do I Log In to My ECS?**

3. Check whether the source ECS can use its primary network interface to communicate with the destination ECS:

   Before configuring policy-based routes, ensure that the source ECS can use its primary network interface to communicate with the destination ECS.

   ping -S *<IP-address-of-the-primary-network-interface-on-the-source-ECS>* *<IP-address-of-the-destination-ECS>*

   In this example, run the following command:

   ping -S 10.0.0.59 10.0.2.12

   If information similar to the following is displayed, the source ECS can use its primary network interface to communicate with the destination ECS.

   ```
   C:\Users\Administrator>ping -S 10.0.0.59 10.0.2.12

   Pinging 10.0.2.12 from 10.0.0.59 with 32 bytes of data:
   Reply from 10.0.2.12: bytes=32 time<1ms TTL=64
   Reply from 10.0.2.12: bytes=32 time<1ms TTL=64
   Reply from 10.0.2.12: bytes=32 time<1ms TTL=64
   Reply from 10.0.2.12: bytes=32 time<1ms TTL=64
   ```

4. Configure a policy-based route for the extended network interface.

   route add -p 0.0.0.0 mask 0.0.0.0 *<subnet-gateway-of-the-extended-network-interface>* **metric** *<route-priority>*

   Configure the parameters as follows:

   - **0.0.0.0/0**: Default route. Do not change it.
   - Subnet gateway of the extended network interface: Enter the IP address collected in **1**.
   - Route priority: Set its value to 261. The priority of the extended network interface must be lower than that of the primary network interface. A larger value indicates a lower priority.

   In this example, run the following command:

   route add -p 0.0.0.0 mask 0.0.0.0 10.0.1.1 metric 261

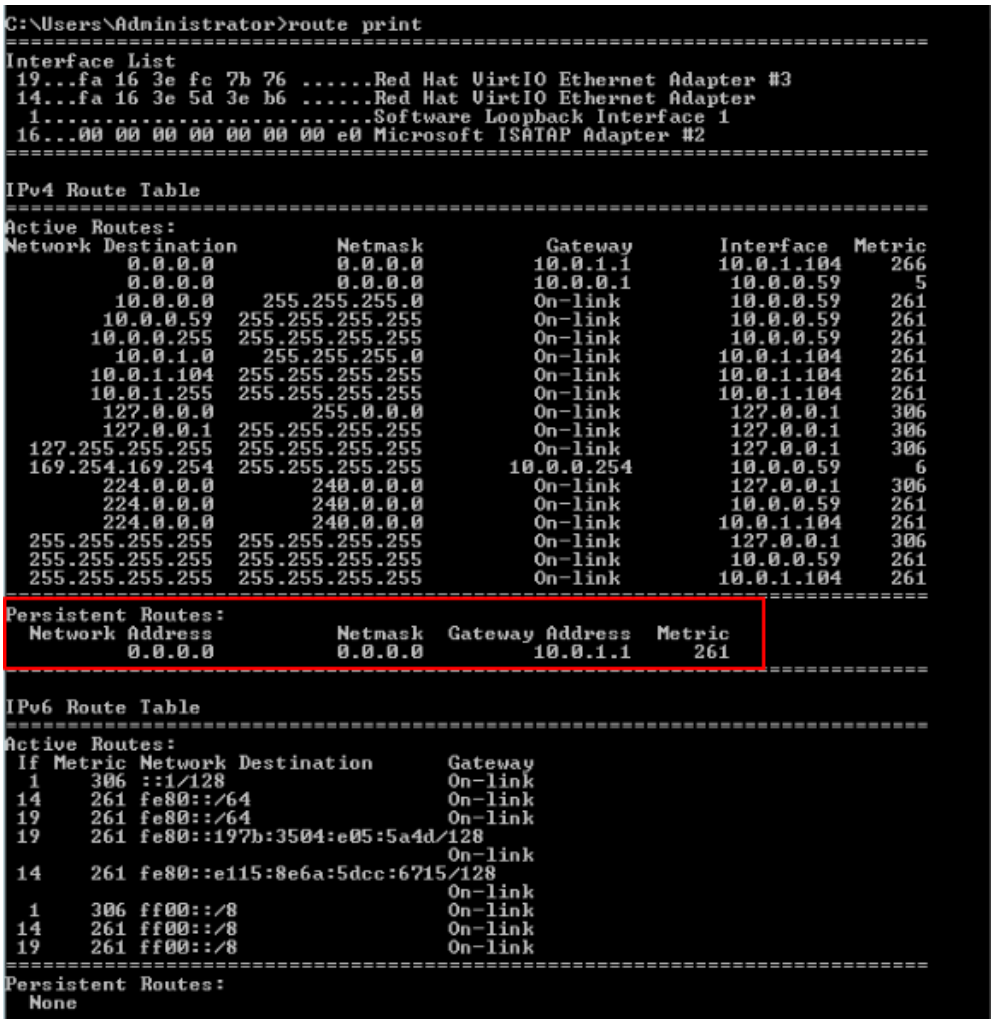
   NOTE

   - The primary network interface already has policy-based routes and you do not need to configure again.
   - If the ECS has multiple extended network interfaces, configure policy-based routes for all extended network interfaces one by one.

5. Check whether the policy-based route is added.

   route print

   If information similar to the following is displayed, the policy-based route has been added. The route is a permanent route and will not be lost after the ECS is restarted.



6. Check whether the source and destination ECSs can communicate with each other.

   ping -S *<IP-address-of-the-primary-network-interface-on-the-source-ECS> <IP-address-of-the-destination-ECS>*

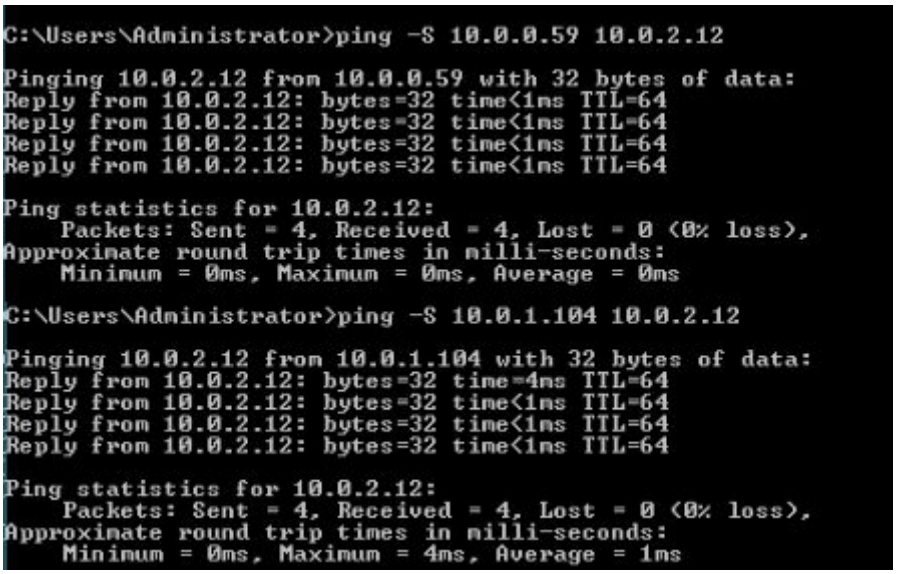
   ping -S *<IP-address-of-the-extended-network-interface-on-the-source-ECS> <IP-address-of-the-destination-ECS>*

   In this example, run the following commands:

   ping -S 10.0.0.59 10.0.2.12

   ping -S 10.0.1.104 10.0.2.12

   If information similar to the following is displayed, both the network interfaces of the source ECS can communicate with the destination ECS.



Configuring IPv6 Policy-based Routes for a Windows ECS

1. Collect the ECS network information required for configuring policy-based routes.

   For details, see **Collecting ECS Network Information**.

   In this example, the network information of the ECS is shown in **Table 5-18**.

   Table 5-18 Windows ECS using IPv6

ECS	Primary Network Interface	Extended Network Interface
Source	IP address: 2407:c080:802:aba:6788:fb94:d71f:8deb	IP address: 2407:c080:802:be6:71c8:42e0:d44e:eeb4
Destination	IP address: 2407:c080:802:be7:c2e6:d99c:b685:c6c8	N/A

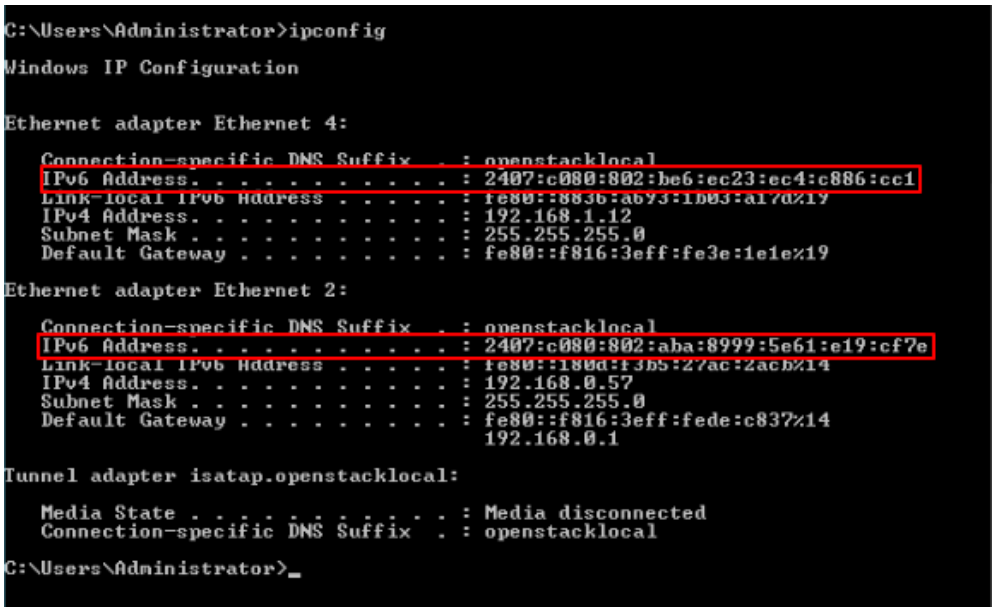
2. Log in to the source ECS.

   For details, see **How Do I Log In to My ECS?**

3. Run the following command to check whether the ECS has IPv6 enabled and has IPv6 addresses:

   ipconfig

   If information similar to the following is displayed, each network interface has an IPv6 address starting with 2407, which indicates that the ECS has been assigned IPv6 addresses.



NOTICE

Perform this step for both the source and destination ECSs to ensure that the ECSs have IPv6 addresses. Otherwise, the ECSs cannot communicate with each other using IPv6 addresses.

ECSs in this example run Windows Server 2012 (64-bit). No additional configuration is required for such ECSs because they can automatically obtain IPv6 addresses. If your ECS cannot automatically obtain IPv6 addresses, see **Dynamically Assigning IPv6 Addresses**.

4. Check whether the source and destination ECSs can communicate with each other.

   ping -6 -S *<IP-address-of-the-primary-network-interface-on-the-source-ECS> <IP-address-of-the-destination-ECS>*

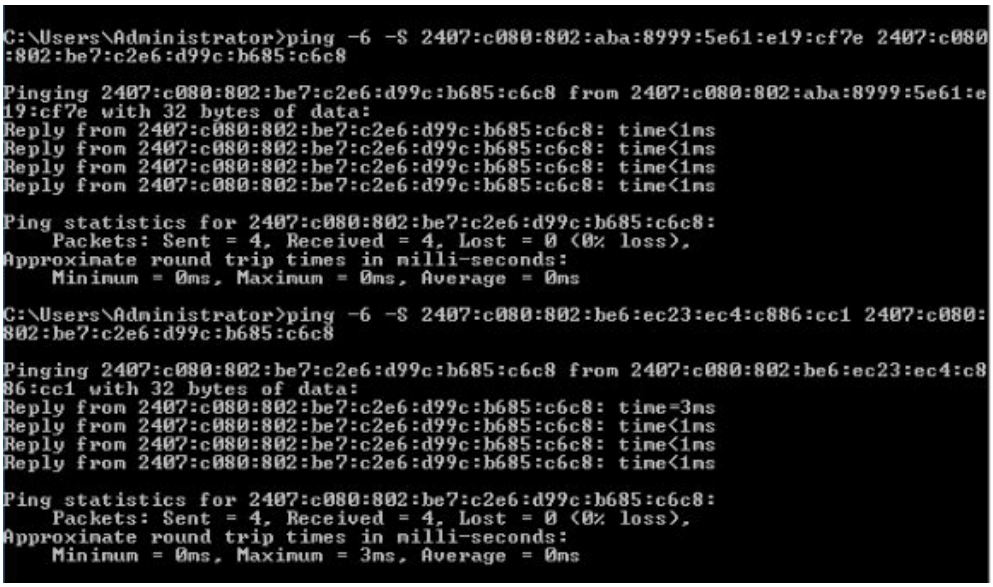
   ping -6 -S *<IP-address-of-the-extended-network-interface-on-the-source-ECS> <IP-address-of-the-destination-ECS>*

   In this example, run the following commands:

   ping -6 -S 2407:c080:802:aba:8999:5e61:e19:cf7e 2407:c080:802:be7:c2e6:d99c:b685:c6c8

   ping -6 -S 2407:c080:802:be6:ec23:ec4:c886:cc1 2407:c080:802:be7:c2e6:d99c:b685:c6c8

   If information similar to the following is displayed, both the network interfaces of the source ECS can communicate with the destination ECS.



NOTICE

ECSs in this example run Windows Server 2012 (64-bit). You do not need to configure policy-based routes for these ECSs because both the network interfaces of such an ECS can communicate with others using IPv6.

6 Access Control

6.1 Access Control Overview

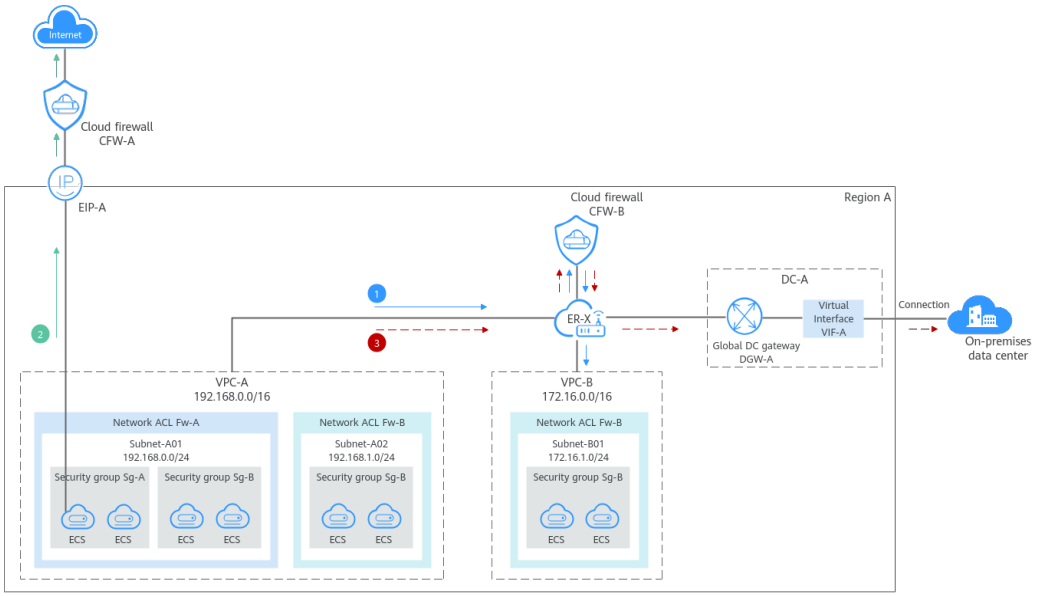
A VPC is your private network on the cloud. You can configure security groups and network ACL rules to ensure the security of instances, such as ECSs, databases, and containers, running in a VPC.

- A security group protects the instances in it.
- A network ACL protects associated subnets and all the resources in the subnets.
- Cloud Firewall filters traffic between VPCs, between VPCs and the Internet, and between VPCs and on-premises data centers, securing access to services. Cloud firewalls offer broader protection compared to security groups and network ACLs.

Figure 6-1 shows how security groups, network ACLs, and cloud firewalls are used. In this figure:

- Security groups Sg-A and Sg-B are used to control the traffic that is entering and leaving ECSs.
- Network ACL Fw-A protects all ECSs in Subnet-A01, while network ACL Fw-B protects all ECSs in Subnet-A02 and Subnet-B01. Network ACLs and security groups are used together to enhance service security.
- Cloud firewalls
  - Filtering traffic between a VPC and the Internet: The ECS accesses the Internet over EIP-A. Cloud firewall CFW-A filters traffic from the ECS to the Internet.
  - Filtering traffic between different VPCs: VPC-A and VPC-B are connected through enterprise router ER-X. Cloud firewall CFW-B filters the traffic between the two VPCs.
  - Filtering traffic between a VPC and an on-premises data center: VPC-A and the on-premises data center are connected through enterprise router ER-X and Direct Connect connection DC-A. Cloud firewall CFW-B filters the traffic from VPC-A to DC-A, and then the filtered traffic is forwarded to the on-premises data center.

Figure 6-1 VPC access control



Differences Between Access Control Options

Table 6-1 provides differences between access control options. You can select one or more as needed.

Table 6-1 Differences between access control options

Item	Security Group	Network ACL	Cloud Firewall
Protection Scope	Protects instances in a security group, such as ECSs, databases, and containers.	Protects subnets and all the instances in the subnets.	Filters traffic between VPCs, between VPCs and the Internet, and between VPCs and on-premises data centers, securing access to services.
Mandatory	Yes. Instances must be added to at least one security group.	No. You can determine whether to associate a subnet with a network ACL based on service requirements.	No. You can determine whether to enable VPC border firewalls based on service requirements.
Billed or Not	No	No	Yes
Stateful	Yes. The response traffic of inbound and outbound requests is allowed to flow to and leave an instance.	Yes. The response traffic of inbound and outbound requests is allowed to flow to and leave a subnet.	Yes. The response traffic of inbound and outbound requests is allowed to flow to and leave the Internet, a VPC, or Direct Connect connection.

Item	Security Group	Network ACL	Cloud Firewall
Rules	Supports both **Allow** and **Deny** rules. • **Allow**: allows the matched traffic to flow in or out of the instances. • **Deny**: denies the matched traffic to flow in or out of the instances.	Supports both **Allow** and **Deny** rules. • **Allow**: allows the matched traffic to flow in or out of the subnet. • **Deny**: denies the matched traffic to flow in or out of the subnet.	Supports both **Allow** and **Block** rules. **Allow**: allows matched traffic to flow into or out of the Internet, a VPC, or direct connection. **Block**: denies matched traffic to flow into or out of the Internet, a VPC, or direct connection.
Rule Packets	Packet filtering based on the 3-tuple (protocol, port, and source/destination)	Packet filtering based on the 5-tuple (protocol, source port, destination port, source, and destination)	Packet filtering based on the 5-tuple (protocol, source port, destination port, source, and destination), domain name, IP geolocation, and Layer 7 protocol

Item	Security Group	Network ACL	Cloud Firewall
Matching Order	If an instance is associated with multiple security groups that have multiple rules: 1. Rules are first matched based on the sequence each security group associated with the instance. Security groups with lower sequence numbers have higher priorities. 2. Rules are then matched by priority in that security group. Rules with lower values have higher priorities than those with higher values. 3. Deny rules take precedence over allow rules if the rules have the same priority.	A subnet can only be associated with one network ACL. If there is more than one rule in a network ACL, they are matched in ascending order, from the lowest to highest rule number.	If there are multiple rules configured for a cloud firewall, the rules are matched based on their priorities. A smaller value indicates a higher priority.

Item	Security Group	Network ACL	Cloud Firewall
Usage	• When creating an instance, for example, an ECS, you must select a security group. If no security group is selected, the ECS will be associated with the default security group. • After creating an instance, you can: – Add or remove the instance to or from a security group on the security group console. – Add or remove the instance to or from a security group on the instance console.	Selecting a network ACL is not allowed when you create a subnet. You must create a network ACL, add inbound and outbound rules, associate subnets with and enable the network ACL. The network ACL then protects the associated subnets and instances in the subnets.	Create a cloud firewall (professional edition) and configure an enterprise router to direct traffic to the cloud firewall. Configure protection rules to allow or block the traffic. CFW provides different features, such as intrusion prevention system (IPS) and antivirus, to filter the allowed traffic.

NOTICE

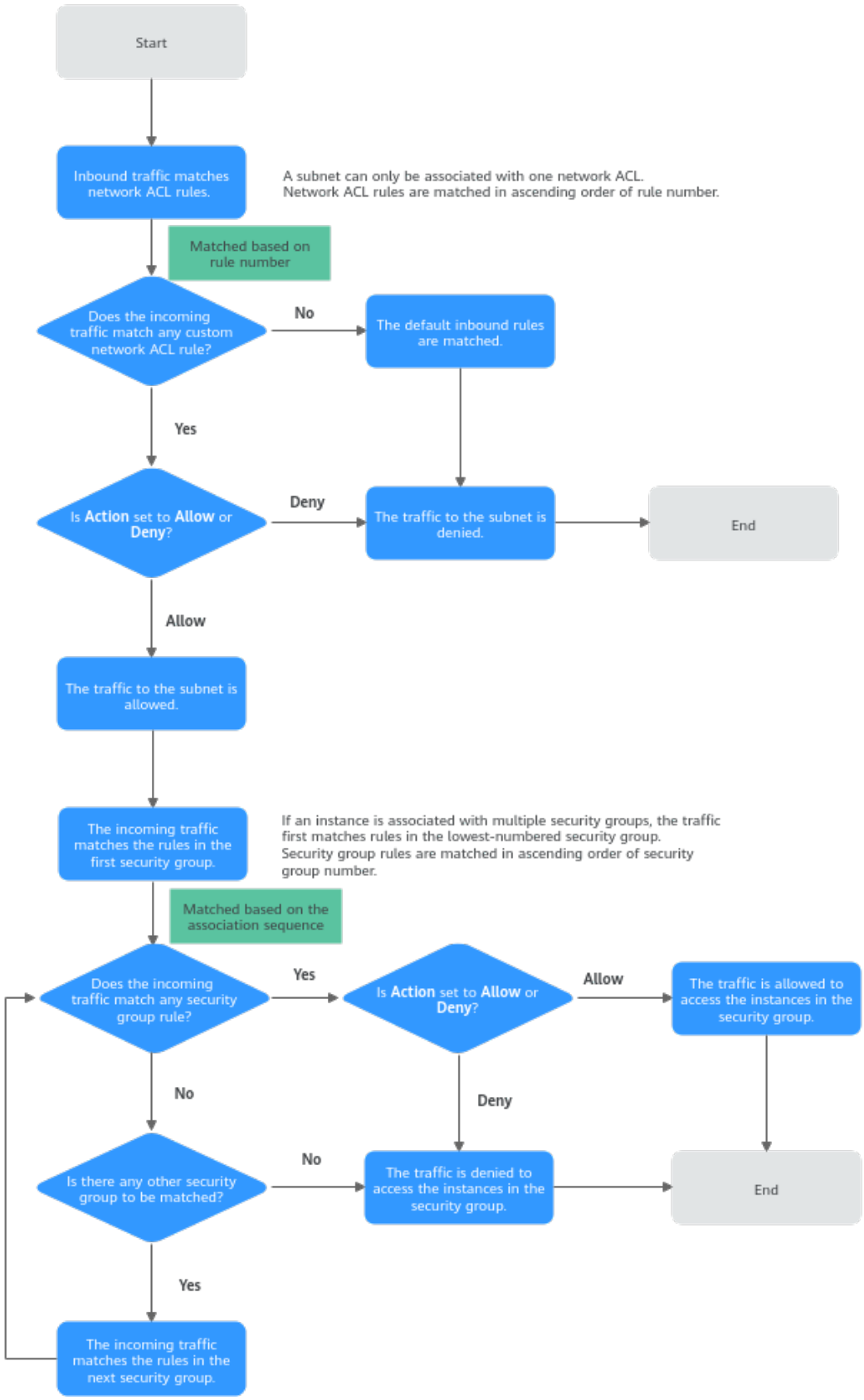
If you need to use advanced protection capabilities (such as IPS, antivirus, and access control based on domain names, geographical locations, and schedules), or your services have high-level protection requirements, you can use **Cloud Firewall (CFW)**.

How Traffic Matches Security Group and Network ACL Rules

If both security group and network ACL rules are configured, traffic matches network ACL rules first and then security group rules. **Figure 6-2** describes how inbound traffic matches security group and network ACL rules.

1. Traffic first matches network ACL rules.
   - If the traffic does not match any rule, the default rule is applied, and traffic to the subnet is denied.
   - If the traffic matches a rule, the rule is applied, which determines where the traffic will go.
     - If **Action** is set to **Deny**, the traffic to the subnet is denied.
     - If **Action** is set to **Allow**, the traffic to the subnet is allowed.
2. The traffic continues to match the security group rules.
   a. If an instance is associated with multiple security groups, the traffic first matches rules in the security group with the lowest sequence number.
      i. If the traffic does not match any rule, it is denied to access the instance.
      ii. If the traffic matches a rule, the rule determines where the traffic will go.
         - If **Action** is set to **Deny**, the traffic is denied to access the instance.
         - If **Action** is set to **Allow**, the traffic is allowed to access the instance.
   b. If the traffic fails to match the rules in the first security group, it continues to match the rules in the second security group.
   c. If the traffic does not match the rules of all security groups, the traffic is denied.

Figure 6-2 How inbound traffic matches security group and network ACL rules



In Figure 6-3, there is a subnet (**Subnet-A**) in **VPC-A**, and two ECSs (**ECS-A** and **ECS-B**) are running in this subnet. To protect your resources in **VPC-A**, you:

- Associate a network ACL (**Fw-A**) with **Subnet-A**. The default rules in **Fw-A** cannot be deleted. Traffic preferentially matches the rules you have configured. Table 6-2 shows some example rules.
- Create a security group **Sg-A** to protect the ECSs. When creating security group **Sg-A**, you can select an existing template. The template comes with some default rules. You can modify or delete default rules, or add rules. For details about security group rules, see Table 6-3.

Figure 6-3 How inbound traffic matches security group and network ACL rules

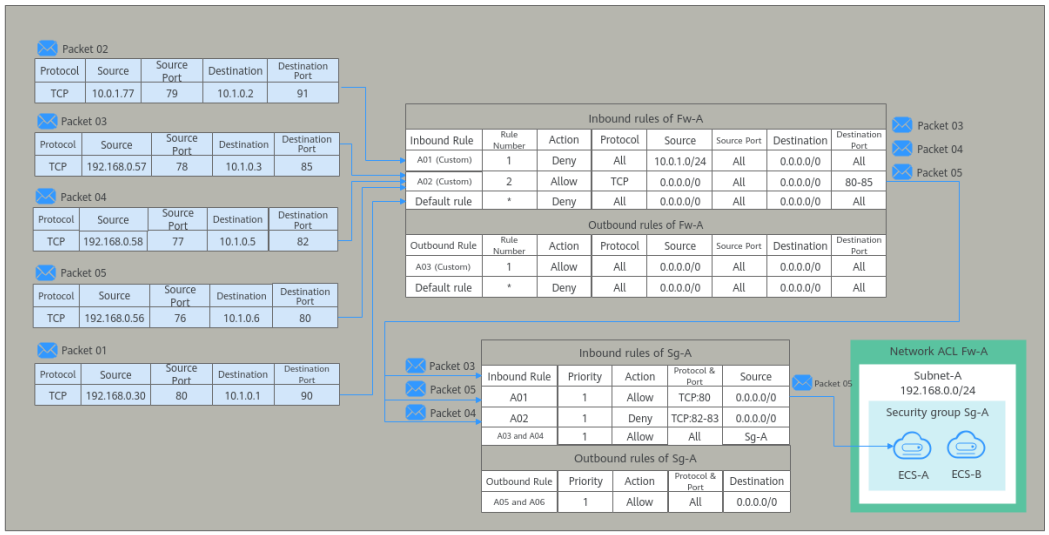


Table 6-2 Rules configured for Fw-A

Direction	Rule Number	Type	Action	Protocol	Source	Source Port Range	Destination	Destination Port Range	Description
Inbound	1	IPv4	Deny	All	10.0.1.0/24	All	0.0.0.0/0	All	Custom rule A01: denies traffic from 10.0.1.0/24 to the subnet.
Inbound	2	IPv4	Allow	TCP	0.0.0.0/0	All	0.0.0.0/0	80-85	Custom rule A02: allows all TCP traffic to the ECS in the subnet over ports 80 to 85.
Inbound	*	--	Deny	All	0.0.0.0/0	All	0.0.0.0/0	All	Default rule: denies all inbound traffic.

Direction	Rule Number	Type	Action	Protocol	Source	Source Port Range	Destination	Destination Port Range	Description
Outbound	1	IPv4	Allow	All	0.0.0.0/0	All	0.0.0.0/0	All	Custom rule A03: allows all outbound traffic.
Outbound	*	--	Deny	All	0.0.0.0/0	All	0.0.0.0/0	All	Default rule: denies all outbound traffic.

Table 6-3 Rules configured for Sg-A

Direction	Priority	Action	Type	Protocol & Port	Source/ Destination	Description
Inbound	1	Allow	IPv4	TCP: 80	Source: 0.0.0.0/0	Rule A01: allows all IPv4 traffic to the ECS over port 80.
Inbound	1	Deny	IPv4	TCP: 82-83	Source: 0.0.0.0/0	Rule A02: denies all IPv4 traffic to the ECS over ports 82 and 83.
Inbound	1	Allow	IPv4	All	Source: current security group (**Sg-A**)	Rule A03: allows the instances in **Sg-A** to communicate with each other over any IPv4 protocol and port.
Inbound	1	Allow	IPv6	All	Source: current security group (**Sg-A**)	Rule A04: allows the instances in **Sg-A** to communicate with each other over any IPv6 protocol and port
Outbound	1	Allow	IPv4	All	Destination: 0.0.0.0/0	Rule A05: allows all traffic from the ECS in the security group to any IPv4 address.
Outbound	1	Allow	IPv6	All	Destination: ::/0	Rule A06: allows all traffic from the ECS in the security group to any IPv6 address.

Based on the preceding scenarios, different inbound packets match rules as follows:

- **Packet 01:** If no custom rules in **Fw-A** are matched, the default rule is applied, denying packet 01 to the subnet.
- **Packet 02:** If custom rule A01 in **Fw-A** is matched, this rule is applied, denying packet 02 to the subnet.
- **Packet 03:** If custom rule A02 in **Fw-A** is matched, this rule is applied, allowing packet 03 to the subnet. Packet 03 continues to match the security group rules. If it does not match any inbound rule in **Sg-A**, packet 03 is denied.
- **Packet 04:** If custom rule A02 in **Fw-A** is matched, this rule is applied, allowing packet 04 to the subnet. Packet 04 continues to match the security group rules. If it matches rule A02 in **Sg-A**, packet 04 is denied.
- **Packet 05:** If custom rule A02 in **Fw-A** is matched, this rule is applied, allowing packet 05 to the subnet. Packet 05 continues to match the security group rules. If it matches rule A01 in **Sg-A**, packet 05 is allowed.

6.2 Security Group

6.2.1 Security Group and Security Group Rule Overview

What Is a Security Group?

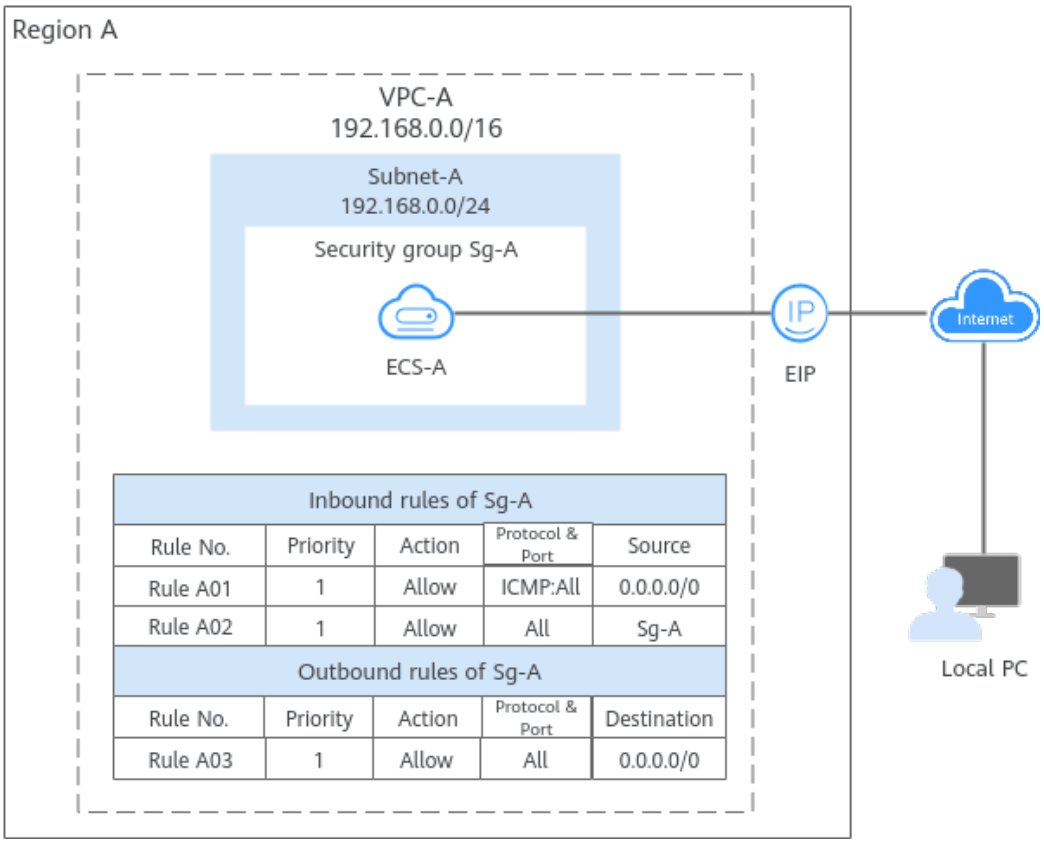
A security group is a collection of access control rules for cloud resources, such as cloud servers, containers, and databases, that have the same security protection requirements and that are mutually trusted. After a security group is created, you can configure access rules that will apply to all cloud resources added to this security group.

When creating an instance (for example, an ECS), you must associate it with a security group. If there are no security groups yet, a **default security group** will be automatically created and associated with the instance. You can also create a security group based on service requirements and associate it with the instance. A cloud resource can be associated with multiple security groups, and traffic to and from the cloud resource is matched by priority in a descending order.

Each security group can have both inbound and outbound rules. You need to specify the source, port, and protocol for each inbound rule and specify the destination, port, and protocol for each outbound rule to control the inbound and outbound traffic to and from the instances in the security group. As shown in **Figure 6-4**, you have a VPC (**VPC-A**) with a subnet (**Subnet-A**) in region A. An ECS (**ECS-A**) is running in **Subnet-A** and associated with security group **Sg-A**.

- Security group **Sg-A** has a custom inbound rule to allow ICMP traffic to **ECS-A** from your PC over all ports. However, the security group does not have rules that allow SSH traffic to **ECS-A** so you cannot remotely log in to **ECS-A** from your PC.
- If **ECS-A** needs to access the Internet through an EIP, the outbound rule of **Sg-A** must allow all traffic from **ECS-A** to the Internet.

Figure 6-4 A security group architecture



NOTE

You can use security groups free of charge.

What Are Security Group Rules?

- A security group has inbound and outbound rules to control traffic that is allowed to reach or leave the instances associated with the security group.
  - Inbound rules: control traffic to the instances in a security group.
  - Outbound rules: control traffic from the instances in a security group to access external networks.
- You can specify protocol, port, source or destination for a security group rule. The following describes key information about a security group.
  - **Action**: **Allow** or **Deny**. If the protocol, port, source or destination of the traffic matches a security group rule, traffic will be allowed or denied.
  - **Priority**: The value ranges from 1 to 100. A smaller value indicates a higher priority. Security group rules are matched by priority and then by action. Deny rules take precedence over allow rules. For more information, see **How Traffic Matches Security Group Rules**.
  - **Type**: IPv4 or IPv6.
  - **Protocol & Port**: network protocol type and port range.
    - Network protocol: The protocol can be TCP, UDP, ICMP, or GRE.

  - Port range: The value ranges from 1 to 65535.
- **Source** or **Destination**: source address of traffic in the inbound direction or destination address of traffic in the outbound direction.

  The source or destination can be an IP address, security group, or IP address group.

  - IP address: a fixed IP address or CIDR block. Both IPv4 and IPv6 addresses are supported, for example, 192.168.10.10/32 (IPv4 address), 192.168.1.0/24 (IPv4 CIDR), or 2407:c080:802:469::/64 (IPv6 CIDR).
  - Security group: If the selected security group and the current security group are in the same region, the traffic is allowed or denied to the private IP addresses of all instances in the selected security group. For example, if there is instance A in security group A and instance B in security group B, and the inbound rule of security group A allows traffic from security group B, traffic is allowed from instance B to instance A.
  - IP address group: If you have multiple IP addresses with the same security requirements, you can add them to an **IP address group** and select this IP address group when you configure a rule, to help you manage them in an easier way.

How Security Groups Work

- Security groups are stateful. If you send a request from your instance and the outbound traffic is allowed, the response traffic for that request is allowed to flow in regardless of inbound security group rules. Similarly, if inbound traffic is allowed, responses to allowed inbound traffic are allowed to flow out, regardless of outbound rules.
- Security groups use connection tracking to track traffic to and from instances. If an inbound rule is modified, the changes immediately take effect for the existing traffic. Changes to outbound security group rules do not affect existing persistent connections and take effect only for new connections.

  If you add, modify, or delete a security group rule, or add or remove an instance to or from a security group, the inbound connections of all instances in the security group will be automatically cleared.
  - The existing inbound persistent connections will be disconnected. All the new connections will match the new rules.
  - The existing outbound persistent connections will not be disconnected, and the original rule will still be applied. All the new connections will match the new rules.

NOTICE

After a persistent connection is disconnected, new connections will not be established immediately until the timeout period of connection tracking expires. For example, after an ICMP persistent connection is disconnected, a new connection will be established and a new rule will be applied when the timeout period (30s) expires.

- The timeout period of connection tracking varies by protocol. The timeout period of a TCP connection in the established state is 600s, and that of an ICMP connection is 30s. For other protocols, if packets are received in both inbound and outbound directions, the connection tracking timeout period is 180s. If packets are received only in one direction, the connection tracking timeout period is 30s.
- The timeout period of TCP connections varies by connection status. The timeout period of a TCP connection in the established state is 600s, and that of a TCP connection in the FIN-WAIT state is 30s.

- Security group rules work like a whitelist. If there are no rules that allow or deny some traffic, the security group denies all traffic to or from the instances in the security group.
  - Inbound rules: If the source of a request matches the source specified in a rule with **Action** set to **Allow**, the request is allowed. For this reason, you do not need to configure a deny rule in the inbound direction.

    The rules in **Table 6-4** ensure that instances in a security group can communicate with each other. Do not delete or modify these rules.
  - Outbound rules: The rules in **Table 6-4** allow all traffic to leave the instances in the security group so that the instances can access any external IP address. If you delete these rules, the instances in the security group cannot access external networks.

Table 6-4 Security group rules

Direction	Action	Type	Protocol & Port	Source/Destination
Inbound	Allow	IPv4	All	Source: current security group
Inbound	Allow	IPv6	All	Source: current security group
Outbound	Allow	IPv4	All	Destination: 0.0.0.0/0
Outbound	Allow	IPv6	All	Destination: ::/0

How Traffic Matches Security Group Rules

An instance can have multiple security groups associated, and a security group can contain multiple security group rules. Security group rules are matched by priority and then by action. Deny rules take precedence over allow rules. The following takes inbound traffic as an example to match security group rules:

1. First, traffic is matched based on the sequence number of security groups. You can adjust the security group sequence. A smaller security group sequence number indicates a higher priority.

   If the sequence number of security group A is 1 and that of security group B is 2, the priority of security group A is higher than that of security group B. Traffic preferentially matches the inbound rules of security group A.

2. Second, traffic is matched based on the priorities and actions of security group rules.

   a. Security group rules are matched by priority first. A smaller value indicates a higher priority.

      If the priority of security group rule A is 1 and that of security group rule B is 2, the priority of security group rule A is higher than that of security group rule B. Therefore, traffic preferentially matches security group rule A.

   b. Deny rules take precedence over allow rules of the same priority.

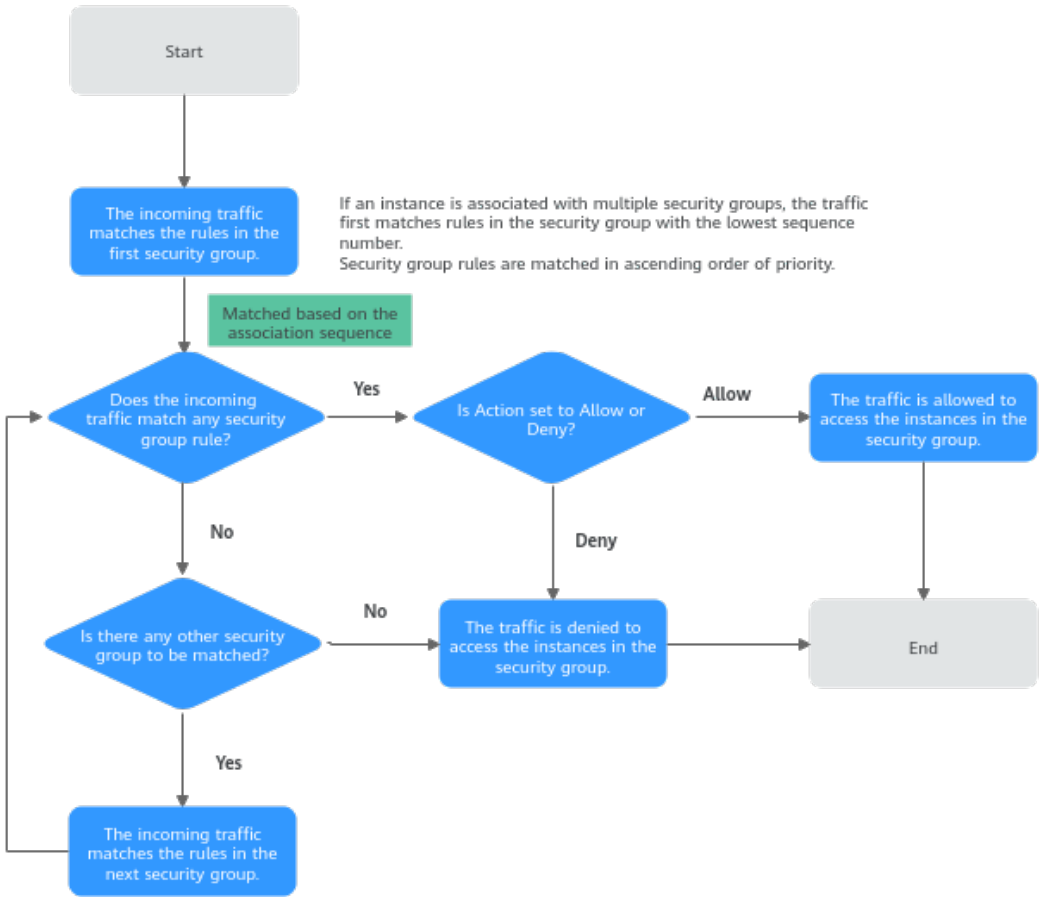
3. Traffic matches all inbound rules of a security group based on the protocol, ports and source.

   – If the traffic matches a rule:

      - With **Action** of **Allow**, the traffic is allowed to access the instances in the security group.
      - With **Action** of **Deny**, the traffic is denied to access the instances in the security group.

   – If the traffic does not match any rule, the traffic is denied to access the instances in the security group.

Figure 6-5 Security group matching sequence



Security Group Examples

You can allow given IP addresses to access instances in a security group, or allow access from another security group to enable instances in different security groups to communicate with each other. You can add security group rules to flexibly control the traffic in and out of a network to ensure network security. The following provides some examples on how security groups can be used.

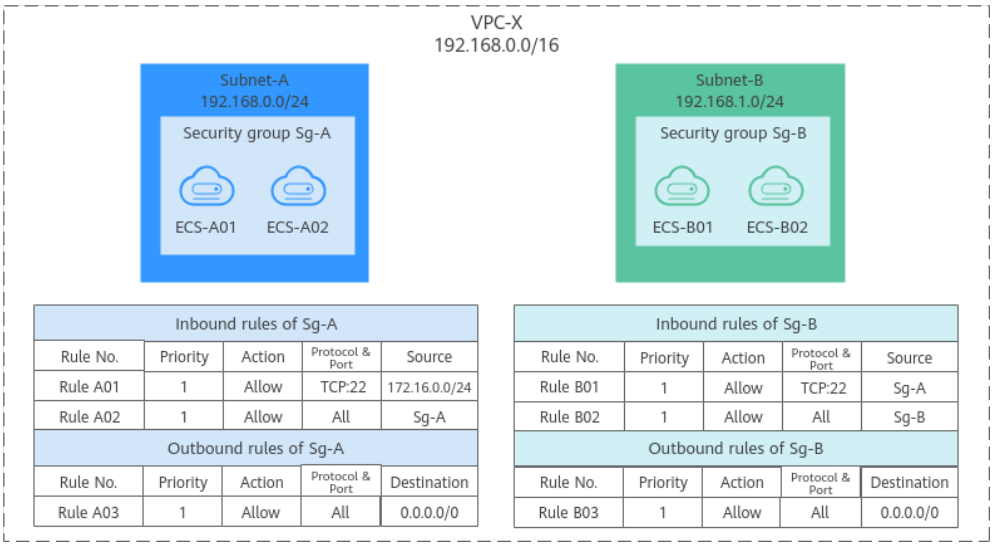
Allowing Traffic from Given IP Addresses or Security Groups

In Figure 6-6, there are two subnets (**Subnet-A** and **Subnet-B**) in **VPC-X**. ECSs in **Subnet-A** are associated with **Sg-A** because these ECSs are used to run the same services and have the same network communication requirements. Similarly, ECSs in **Subnet-B** are associated with security group **Sg-B**.

- Inbound rule A01 of **Sg-A** allows traffic from IP addresses in **172.16.0.0/24** to access SSH port 22 of the ECSs in **Sg-A** for remotely logging in to these ECSs.
- Inbound rule A02 of **Sg-A** allows the ECSs in this security group to communicate with each other using any protocol and port.
- Inbound rule B01 of **Sg-B** allows the ECSs in **Sg-A** to access SSH port 22 of the ECSs in **Sg-B** for remotely logging in to the ECSs in **Subnet-B**.
- Inbound rule B02 of **Sg-B** allows the ECSs in this security group to communicate with each other using any protocol and port.

- The outbound rules of both security groups allow all traffic from the ECSs in the security groups.

Figure 6-6 Allowing traffic from given IP addresses and security groups



Inbound rules of Sg-A

Rule No.	Priority	Action	Protocol & Port	Source
Rule A01	1	Allow	TCP:22	172.16.0.0/24
Rule A02	1	Allow	All	Sg-A

Outbound rules of Sg-A

Rule No.	Priority	Action	Protocol & Port	Destination
Rule A03	1	Allow	All	0.0.0.0/0

Inbound rules of Sg-B

Rule No.	Priority	Action	Protocol & Port	Source
Rule B01	1	Allow	TCP:22	Sg-A
Rule B02	1	Allow	All	Sg-B

Outbound rules of Sg-B

Rule No.	Priority	Action	Protocol & Port	Destination
Rule B03	1	Allow	All	0.0.0.0/0

NOTE

Security Group Examples lists more security group rule configuration examples.

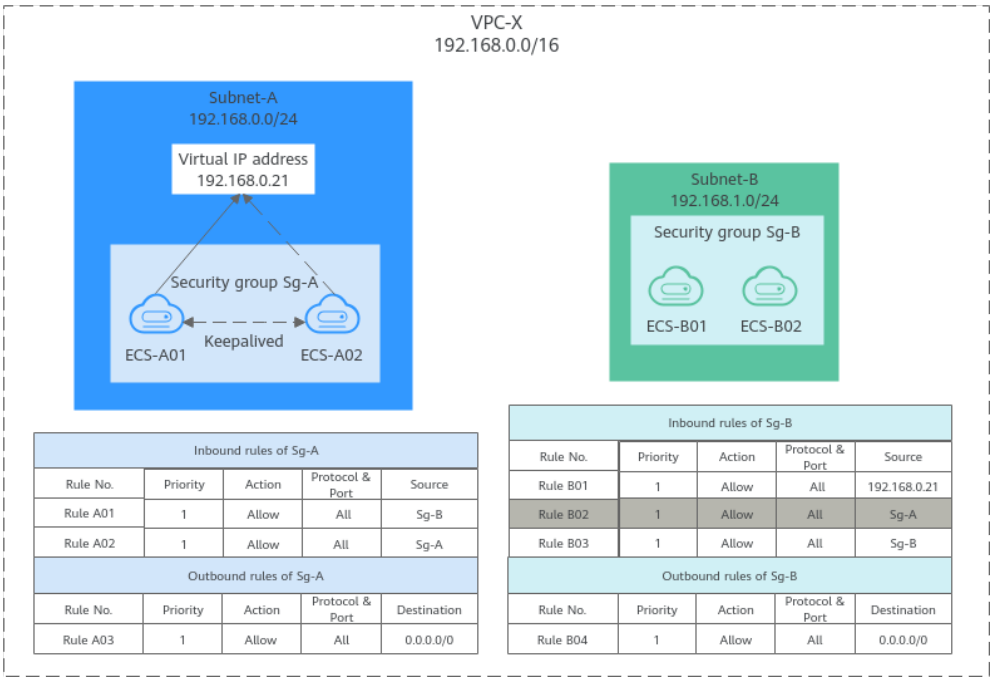
Allowing Traffic from a Virtual IP Address

In **Figure 6-7**, ECSs in **Subnet-A** and **Subnet-B** are connected by a virtual IP address. If you set the source of inbound rule to the security group associated with the peer ECS, the ECSs in the two security groups cannot communicate with each other, because they are connected by a virtual IP address. You need to set the source to the private IP address or subnet CIDR block of the virtual IP address.

In **Figure 6-7**, **VPC-X** has two subnets: **Subnet-A** and **Subnet-B**. ECSs in **Subnet-A** are associated with security group **Sg-A**, and ECSs in **Subnet-B** are associated with security group **Sg-B**. **ECS-A01** and **ECS-A02** work in active/standby pair, forming a Keepalived HA cluster. The ECSs use virtual IP address **192.168.0.21** to communicate with external networks.

- Inbound rule A01 of **Sg-A** allows ECSs in **Sg-B** to access ECSs in **Sg-A** using any protocol over any port.
- **Sg-B** has the following inbound rules:
  - Rule B02: Allows ECSs in **Sg-A** to use private IP addresses to access ECSs in **Sg-B**. However, in this networking, ECSs in **Sg-A** are supposed to communicate with ECSs in **Sg-B** through virtual IP address **192.168.0.21**. However, rule B02 does not allow traffic from this virtual IP address.
  - Rule B01: Allows traffic from virtual IP address **192.168.0.21** to ECSs in **Sg-B** using any protocol over port. In this networking, you can also set the source to **192.168.0.0/24**, the CIDR block of **Subnet-A**.

Figure 6-7 Allowing traffic from a virtual IP address



NOTE

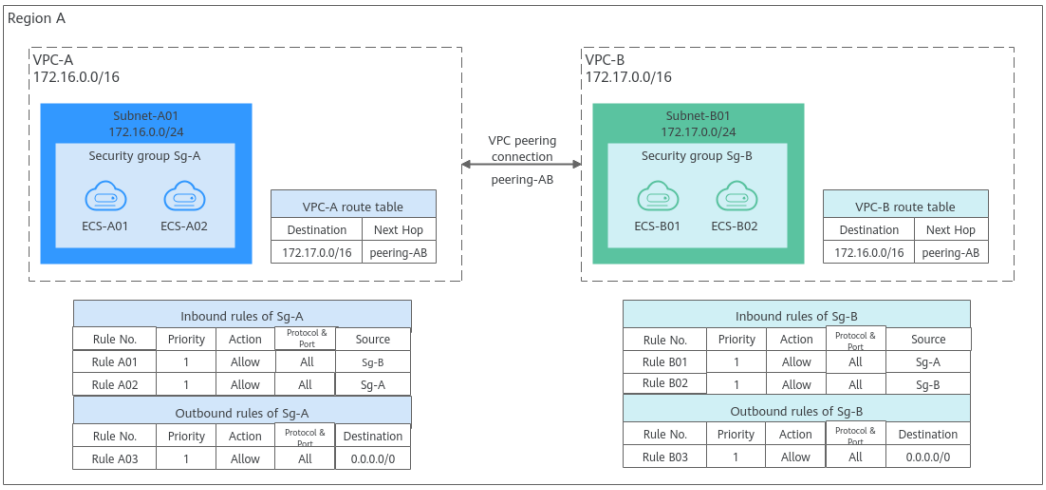
Security Group Examples lists more security group rule configuration examples.

Allowing Communications Between Instances in Two VPCs Connected by a VPC Peering Connection

In **Figure 6-8**, **VPC-A** and **VPC-B** in region A are connected by VPC peering connection **peering-AB**. After routes are configured for the VPC peering connection, **Subnet-A** and **Subnet-B** can communicate with each other. However, the ECSs in the two subnets are associated with different security groups. To allow ECSs in **Sg-A** and **Sg-B** to communicate with each other, you can add the following rules:

- Rule A01 with **Source** to **Sg-B** to allow ECSs in **Sg-B** to access ECSs in **Sg-A**.
- Rule B01 with **Source** to **Sg-A** to allow ECSs in **Sg-A** to access ECSs in **Sg-B**.

Figure 6-8 Allowing communications between ECSs in two VPCs connected by a VPC peering connection



NOTE

Security Group Examples lists more security group rule configuration examples.

Security Group Configuration Process

Figure 6-9 Process of using a security group

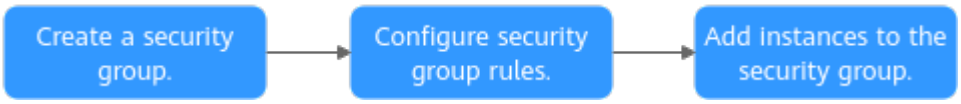


Table 6-5 Security group configuration process description

No.	Step	Description	Reference
1	Create a security group.	When creating a security group, you can use the preset rules. For details about the preset security group rules, see **Table 6-23**.	**Creating a Security Group**
2	Configure security group rules.	After a security group is created, if its rules cannot meet your service requirements, you can add new rules to the security group or modify original rules.	**Adding a Security Group Rule** **Fast-Adding Security Group Rules**
3	Add instances to the security group.	When you create an instance, the system automatically adds the instance to a security group for protection. If one security group cannot meet your requirements, you can add an instance to multiple security groups.	**Adding an Instance to or Removing an Instance from a Security Group**

Constraints

- For better network performance, you are advised to associate an instance with no more than five security groups.
- A security group can have no more than 6,000 instances associated, or its performance will deteriorate.
- For inbound security group rules, the sum of the rules with **Source** set to **Security group**, of the rules with **Source** set to **IP address group**, and of the rules with inconsecutive ports, cannot exceed 120. If there are both IPv4 and IPv6 security group rules, up to 120 rules can be added for each type.

  The limits on outbound security group rules are the same as those on inbound rules.

  For example, to add inbound IPv4 rules to a security group (Sg-A), you can refer to **Table 6-6** for rules that meet the restrictions. Of these rules, rule A02 uses inconsecutive ports (TCP: 22,25,27) and security group Sg-B as the source. In this case, only one quota is occupied.

  Table 6-6 Inbound security group rules

Rule No.	Action	Type	Protocol & Port	Source
Rule A01	Allow	IPv4	All	Current security group: Sg-A
Rule A02	Allow	IPv4	**TCP: 22,25,27**	**Another security group: Sg-B**
Rule A03	Allow	IPv4	TCP: 80-82	IP address group: ipGroup-A
Rule A04	Allow	IPv4	TCP: 22-24,25	IP address: 192.168.0.0/16

- If you specify an IP address group or inconsecutive ports for a security group rule, the rule is only applied for certain ECSs. For details, see **Table 6-7**.

  Table 6-7 Scenarios that security group rules do not take effect

Rule Configuration	ECS Type
Source or **Destination** is set to **IP address group**.	The following x86 ECS types are not supported: • General computing (S1, C1, and C2 ECSs) • Memory-optimized (M1 ECSs) • High-performance computing (H1 ECSs) • Disk-intensive (D1 ECSs) • GPU-accelerated (G1 and G2 ECSs) • Large-memory (E1, E2, and ET2 ECSs)

Rule Configuration	ECS Type
Port is set to non-consecutive ports.	The following x86 ECS types are not supported: • General computing (S1, C1, and C2 ECSs) • Memory-optimized (M1 ECSs) • High-performance computing (H1 ECSs) • Disk-intensive (D1 ECSs) • GPU-accelerated (G1 and G2 ECSs) • Large-memory (E1, E2, and ET2 ECSs)
	All Kunpeng ECS flavors do not support inconsecutive ports. If you use inconsecutive port numbers in a security group rule of a Kunpeng ECS, this rule and rules configured after this one do not take effect. If you configure security group rule A with inconsecutive ports **22,24** and then configure security group rule B with port 9096, both rule A and rule B do not take effect.

NOTE

- For details about x86 ECSs, see **ECS Specifications (x86)**.
- For details about Kunpeng ECSs, see **ECS Specifications (Kunpeng)**.

- Traffic from load balancers is not restricted by network ACL and security group rules if:

  Transfer Client IP Address is enabled for the listener of a load balancer.

  The load balancer can still forward traffic to backend servers, even if there is a rule that denies traffic from the load balancer to the backend servers.

Recommendations

- Instances in a security group deny all external access requests by default, but you can add rules to allow specific requests.
- When adding a security group rule, grant the minimum permissions possible. For example, if remote login to an ECS over port 22 is allowed, only allow specific IP addresses to log in to the ECS. Do not use 0.0.0.0/0 (all IP addresses).
- Keep your configurations simple. There should be a different reason for each security group. If you use the same security group for all your different instances, the rules in the security group will likely be redundant and complex. It will make it much harder to maintain and manage.
- You can add instances to different security groups based on their functions. For example, if you want to provide website services accessible from the Internet, you can add the web servers to a security group configured for that specific purpose and only allow external access over specific ports, such as 80

and 443. By default, other external access requests are denied. Do not run internal services, such as MySQL or Redis, on web servers that provide services accessible from the Internet. Deploy internal services on servers that do not need to connect to the Internet and associate these servers with security groups specifically configured for that purpose.

- If you have multiple IP addresses with the same security requirements, you can add them to an IP address group and select this IP address group when you configure a rule, to help you manage them in an easier way. When an IP address changes, you only need to change the IP address in the IP address group. Then, the rules in the IP address group change accordingly. You do not need to modify the rules in the security group one by one. This simplifies security group management and improves efficiency. For details, see **Using IP Address Groups to Reduce the Number of Security Group Rules**.
- Do not directly modify security group rules for active services. Before you modify security group rules used by a service, you can clone the security group and modify the security group rules in the test environment to ensure that the modified rules work. For details, see **Cloning a Security Group**.
- After you add instances to or modify rules of a security group, the security group rules are applied automatically. There is no need to restart the instances.

  If a security group rule does not take effect after being configured, see **Why Are My Security Group Rules Not Applied?**

6.2.2 Default Security Groups

When creating an instance, you must associate it with a security group. If no security group has been created yet, a default security group will be created and associated with the instance. Note the following when using default security groups:

- The default security group name is **default**. It is recommended that you do not change the name of the default security group in order to distinguish it from custom security groups.
- You cannot delete the default security group, but you can modify its rules or add rules to it.
- The default security group denies all external requests. To allow access to an instance associated with this security group, you can add rules to allow access over given ports by referring to **Remotely Logging In to an ECS from a Local Server**.
- If your service has different security requirements on instances for different purposes, you can create security groups and associate these instances with different security groups based on their purposes.

  NOTE

  Security groups are free of charge.

Default Security Group Rules

Note the following when using default security group rules:

- Inbound rules control incoming traffic to instances in the default security group. The instances can only communicate with each other but cannot be accessed from external networks.

- Outbound rules allow all traffic from the instances in the default security group to external networks.

Figure 6-10 Default security group

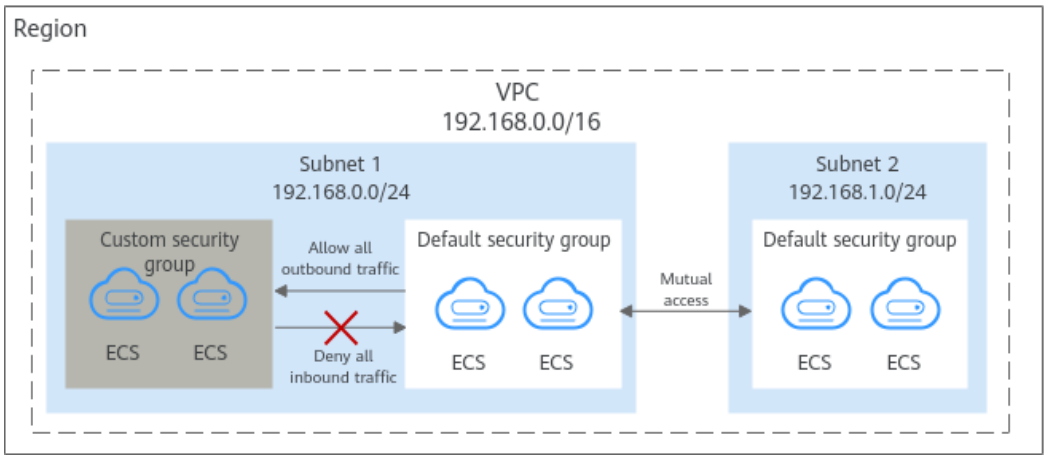


Table 6-8 describes the default rules for the default security group.

Table 6-8 Default security group rules

Direction	Action	Type	Protocol & Port	Source/ Destination	Description
Inbound	Allow	IPv4	All	Source: default security group (default)	Allows IPv4 instances in the security group to communicate with each other using any protocol over any port.
Inbound	Allow	IPv6	All	Source: default security group (default)	Allows IPv6 instances in the security group to communicate with each other using any protocol over any port.
Outbound	Allow	IPv4	All	Destination: 0.0.0.0/0	Allows all traffic from the instances in the security group to any IPv4 address over any port.
Outbound	Allow	IPv6	All	Destination: ::/0	Allows all traffic from the instances in the security group to any IPv6 address over any port.

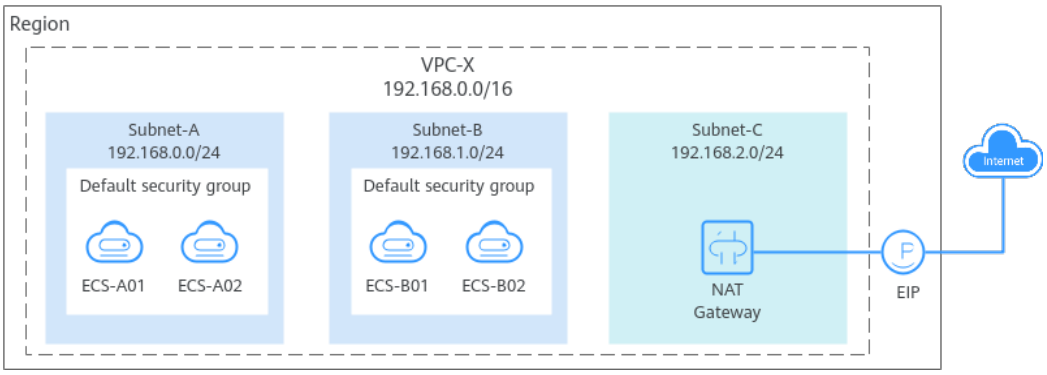
A Default Security Group Example

As shown in Figure 6-11, **VPC-X** has three subnets: **Subnet-A**, **Subnet-B**, and **Subnet-C**. ECSs in **Subnet-A** and **Subnet-B** have been associated with the default security group. The default security group allows the instances in the security

group to communicate with each other and denies all external requests. So, the four ECSs (**ECS-A01**, **ECS-A02**, **ECS-B01**, and **ECS-B02**) can communicate with each other, but they cannot receive traffic from the NAT gateway.

To allow traffic from the NAT gateway, you need to add rules to the default security group or create a security group and associate it with the instances.

Figure 6-11 Use cases



6.2.3 Security Group Examples

When creating instances, such as cloud servers, containers, and databases, in a VPC subnet, you can use the default security group or create a custom security group, and then add inbound and outbound rules to control traffic from and to the instances in the security group. Here are some common security group configuration examples:

- **Remotely Logging In to an ECS from a Local Server**
- **Remotely Connecting to an ECS from a Local Server to Upload or Download Files over FTP**
- **Setting Up a Website on an ECS to Provide Internet-Accessible Services**
- **Using ping Command to Verify Network Connectivity**
- **Enabling Communications Between Instances in Different Security Groups**
- **Allowing External Instances to Access the Database Deployed on an ECS**
- **Allowing ECSs to Access Only Specific External Websites**

NOTICE

If your security group rules are not working right, **submit a service ticket**.

Precautions

Note the following before configuring security group rules:

- Instances associated with different security groups are isolated from each other by default.
- Generally, a security group denies all external requests by default, while allowing instances in it to communicate with each other.

If required, you can add inbound rules to allow specific traffic to access the instances in the security group.

- If the source is set to 0.0.0.0/0 or::/0, then the access from all external IP addresses are either allowed or denied, depending on if the action is **Allow** or **Deny**. If the access is allowed, exposing **high-risk ports**, such as port 22, 3389, or 8848, to the public network will leave your instances vulnerable to network intrusions, service interruptions, data leakage, or ransomware attacks. You should only configure trusted IP addresses for the security group rule.
- By default, outbound security group rules allow all requests from the instances in the security group to access external resources.

  If outbound rules are deleted, the instances in the security group cannot communicate with external resources. To allow outbound traffic, you need to add outbound rules by referring to **Table 6-9**.

Table 6-9 Default outbound rules in a security group

Direction	Priority	Action	Type	Protocol & Port	Destination	Description
Outbound	1	Allow	IPv4	All	0.0.0.0/0	Allows the instances in the security group to access any IPv4 address over any port.
Outbound	1	Allow	IPv6	All	::/0	Allows the instances in the security group to access any IPv6 address over any port.

Remotely Logging In to an ECS from a Local Server

A security group denies all external requests by default. To remotely log in to an ECS in a security group from a local server, add an inbound rule based on the OS running on the ECS.

- To remotely log in to a Linux ECS using SSH, enable port 22. For details, see **Table 6-10**.
- To remotely log in to a Windows ECS using RDP, enable port 3389. For details, see **Table 6-11**.

Table 6-10 Remotely logging in to a Linux ECS using SSH

Direction	Priority	Action	Type	Protocol & Port	Source
Inbound	1	Allow	IPv4	TCP: 22	IP address: 0.0.0.0/0

Table 6-11 Remotely logging in to a Windows ECS using RDP

Direction	Priority	Action	Type	Protocol & Port	Source
Inbound	1	Allow	IPv4	TCP: 3389	IP address: 0.0.0.0/0

NOTICE

If the source is set to 0.0.0.0/0, all external IP addresses are allowed to remotely log in to the ECS. To ensure network security and prevent service interruptions caused by network intrusions, set the source to a trusted IP address. For details, see **Table 6-12**.

Table 6-12 Remotely logging in to an ECS using a trusted IP address

ECS Type	Direction	Priority	Action	Type	Protocol & Port	Source
Linux ECS	Inbound	1	Allow	IPv4	TCP: 22	IP address: 192.168.0.0/24
Windows ECS	Inbound	1	Allow	IPv4	TCP: 3389	IP address: 10.10.0.0/24

Remotely Connecting to an ECS from a Local Server to Upload or Download Files over FTP

By default, a security group denies all external requests. If you need to remotely connect to an ECS from a local server to upload or download files, you need to enable FTP ports 20 and 21.

Table 6-13 Remotely connecting to an ECS from any server to upload or download files over FTP

Direction	Priority	Action	Type	Protocol & Port	Source
Inbound	1	Allow	IPv4	TCP: 20-21	IP address: 0.0.0.0/0

NOTICE

- If the source is set to 0.0.0.0/0, all external IP addresses are allowed to remotely log in to the ECS to upload or download files. To ensure network security and prevent service interruptions caused by network intrusions, set the source to a trusted IP address. For details, see **Table 6-14**.
- You must first install the FTP server program on the ECSs and then check whether ports 20 and 21 are working properly.

Table 6-14 Remotely connecting to an ECS from a trusted server to upload or download files

Direction	Priority	Action	Type	Protocol & Port	Source
Inbound	1	Allow	IPv4	TCP: 20-21	IP address: 192.168.0.0/24

Setting Up a Website on an ECS to Provide Internet-Accessible Services

A security group denies all external requests by default. If you set up a website on an ECS to allow access from the Internet, you need to add an inbound rule to the ECS security group to allow access over specific ports, such as HTTP (80) and HTTPS (443).

Table 6-15 Setting up a website on an ECS to provide services internet-accessible services

Direction	Priority	Action	Type	Protocol & Port	Source
Inbound	1	Allow	IPv4	TCP: 80	IP address: 0.0.0.0/0
Inbound	1	Allow	IPv4	TCP: 443	IP address: 0.0.0.0/0

Using ping Command to Verify Network Connectivity

Ping works by sending an Internet Control Message Protocol (ICMP) Echo Request. To ping an ECS from your PC to verify the network connectivity, you need to add an inbound rule to the security group of the ECS to allow ICMP traffic.

Table 6-16 Using **ping** command to verify network connectivity

Direction	Priority	Action	Type	Protocol & Port	Source
Inbound	1	Allow	IPv4	ICMP: All	IP address: 0.0.0.0/0

Direction	Priority	Action	Type	Protocol & Port	Source
Inbound	1	Allow	IPv6	ICMP: All	IP address: ::/0

Enabling Communications Between Instances in Different Security Groups

Instances in the same VPC but in different security groups cannot communicate with each other. If you want ECSs in security group **sg-A** to access MySQL databases in security group **sg-B**, you need to add an inbound rule to security group **sg-B** to allow access from ECSs in security group **sg-A**.

Table 6-17 Enabling communications between instances in different security groups

Direction	Priority	Action	Type	Protocol & Port	Source
Inbound	1	Allow	IPv4	TCP: 3306	Security group: sg-A

NOTICE

If you use an intermediate network instance to forward traffic between instances in different subnets, setting the source of inbound rule to the security group associated with the peer instance does not allow the instances to communicate with each other. To enable communications, set the source to the private IP address or subnet CIDR block of the intermediate network instance. For example, to connect ECSs in **Subnet-A** and **Subnet-B** as described in the second security group example in **Security Group Examples**, set the source of inbound rule to the virtual IP address.

Allowing External Instances to Access the Database Deployed on an ECS

A security group denies all external requests by default. If you have deployed a database on an ECS and want the database to be accessed from external instances on a private network, you need to add an inbound rule to the security group of the ECS to allow access over corresponding ports. Here are some common ports for databases:

- MySQL: port 3306
- Oracle: port 1521
- MS SQL: port 1433
- PostgreSQL: port 5432
- Redis: port 6379

Table 6-18 Allowing external instances to access the database deployed on an ECS

Direction	Priority	Action	Type	Protocol & Port	Source	Description
Inbound	1	Allow	IPv4	TCP: 3306	Security group: sg-A	Allows the ECSs in security group **sg-A** to access the MySQL database.
Inbound	1	Allow	IPv4	TCP: 1521	Security group: sg-B	Allows the ECSs in security group **sg-B** to access the Oracle database.
Inbound	1	Allow	IPv4	TCP: 1433	IP address: 172.16.3.21/32	Allows the ECS whose private IP address is 172.16.3.21 to access the MS SQL database.
Inbound	1	Allow	IPv4	TCP: 5432	IP address: 192.168.0.0/24	Allows ECSs whose private IP addresses are in the 192.168.0.0/24 network to access the PostgreSQL database.
Inbound	1	Allow	IPv4	TCP: 6379	IP address group: ipGroup-A	Allows ECSs whose private IP addresses are in IP address group **ipGroup-A** to access the Redis database.

NOTICE

In this example, the source IP addresses are for reference only. Replace them with actual IP addresses.

Allowing ECSs to Access Only Specific External Websites

By default, a security group allows all outbound traffic. **Table 6-20** lists the default outbound rules. If you want to allow ECSs to access only specific websites, configure the security group as follows:

1. Add outbound rules to only allow traffic over specific ports to specific IP addresses.

Table 6-19 Allowing ECSs to access only specific external websites

Direction	Priority	Action	Type	Protocol & Port	Destination	Description
Outbound	1	Allow	IPv4	TCP: 80	IP address: 132.15.XX.XX	Allows ECSs in the security group to access the external website at http://132.15.XX.XX:80.
Outbound	1	Allow	IPv4	TCP: 443	IP address: 145.117.XX.XX	Allows ECSs in the security group to access the external website at https://145.117.XX.XX:443.

2. Delete the default outbound rules that allow all traffic.

Table 6-20 Default outbound rules in a security group

Direction	Priority	Action	Type	Protocol & Port	Destination	Description
Outbound	1	Allow	IPv4	All	0.0.0.0/0	Allows the instances in the security group to access any IPv4 address over any port.
Outbound	1	Allow	IPv6	All	::/0	Allows the instances in the security group to access any IPv6 address over any port.

6.2.4 Common ECS Ports

When adding a security group rule, you must specify a port or port range for communications. Traffic is then allowed or denied if traffic matches this rule. Suppose a client requests to remotely log in to an ECS using SSH. When the request reaches the security group, the IP address and port of the client will be checked. If the IP address and the port match the allow rules in the security group, the request is allowed.

Table 6-21 lists some high-risk ports that are blocked by default. Even if you have added a security group rule to allow access over these ports, traffic over these ports in restricted regions is still denied. In this case, do not use these high-risk ports for your services.

Table 6-21 High-risk ports

Protocol	Port
TCP	42, 135, 137, 138, 139, 444, 445, 593, 1025, 1068, 1433, 1434, 3127, 3128, 3129, 3130, 4444, 4789, 5554, 5800, 5900, 8998, 9995, and 9996
UDP	135~139 1026 1027 1028 1068 1433 1434 4789 5554 9995 9996

Common Ports

Table 6-22 lists the common ports used by ECSs. You can configure security group rules to allow traffic to and from specified ECS ports. For details, see **Adding a Security Group Rule**. For more information about requirements for Windows, see **Service overview and network port requirements for Windows**.

Table 6-22 Common ports used by ECSs

Port	Protocol	Description
21	FTP	Used by FTP services for uploading and downloading files. For configuration examples, see **Remotely Connecting to an ECS from a Local Server to Upload or Download Files over FTP**.
22	SSH	Used to remotely connect to Linux ECSs. For configuration examples, see **Remotely Logging In to an ECS from a Local Server**. For details about how to log in to a Linux ECS, see **Linux ECS Login Overview**.
23	Telnet	Used to remotely log in to ECSs.
25	SMTP	Used to send emails. For security purposes, TCP port 25 is disabled in the outbound direction by default. For details about how to open the port, see **Why Is Outbound Access Through TCP Port 25 Restricted?**
80	HTTP	Used to access websites over HTTP. For configuration examples, see **Setting Up a Website on an ECS to Provide Internet-Accessible Services**.
110	POP3	Used to receive emails using Post Office Protocol version 3 (POP3).
143	IMAP	Used to receive emails using Internet Message Access Protocol (IMAP).
443	HTTPS	Used to access websites over HTTPS. For configuration examples, see **Setting Up a Website on an ECS to Provide Internet-Accessible Services**.

Port	Protocol	Description
1433	SQL Server	A TCP port of the SQL Server for providing services. For configuration examples, see **Allowing External Instances to Access the Database Deployed on an ECS**.
1434	SQL Server	A UDP port of the SQL Server for returning the TCP/IP port number used by the SQL Server. For configuration examples, see **Allowing External Instances to Access the Database Deployed on an ECS**.
1521	Oracle	Used for Oracle database communications. This port must be enabled on the ECSs where Oracle SQL Server is deployed. For configuration examples, see **Allowing External Instances to Access the Database Deployed on an ECS**.
3306	MySQL	Used by MySQL databases to provide services. For configuration examples, see **Allowing External Instances to Access the Database Deployed on an ECS**.
3389	Windows Server Remote Desktop Services	Used to connect to Windows ECSs. For configuration examples, see **Remotely Logging In to an ECS from a Local Server**. For details about how to log in to a Windows ECS, see **Windows ECS Login Overview**.
8080	Proxy	Used by the WWW proxy service for web browsing, like port 80. If you use port 8080, you need to add **:8080** after the IP address when you visit a website or use a proxy server. If Apache Tomcat is installed, its default service port is 8080.
137, 138, and 139	NetBIOS	Used for Windows files, printer sharing, and Samba. • Ports 137 and 138: UDP ports that are used when files are transferred using Network Neighborhood (My Network Places). • Port 139: Connections from this port try to access the NetBIOS/SMB service.

6.2.5 Managing a Security Group

6.2.5.1 Creating a Security Group

Scenarios

A security group consists of inbound and outbound rules. You can add security group rules to allow or deny the traffic to reach and leave the instances (such as ECSs) in the security group.

When creating an instance (for example, an ECS), you must associate it with a security group. If no security group has been created yet, a **default security group**

will be created and associated with the instance. You can also create a security group and add inbound and outbound rules to allow specific traffic. For more information about security groups and rules, see **Security Group and Security Group Rule Overview**.

Security Group Templates

When creating a security group, you can select preset rules. The preset rules have preconfigured inbound and outbound rules. You can select rules as needed. **Table 6-23** describes the preset rules.

Table 6-23 Security group rules

Template	Direction	Protocol/ Port/ Type	Source/ Destination	Description	Scenario
General-purpose web server	Inbound	TCP: 22 (IPv4)	0.0.0.0/0	Allows all IPv4 addresses to access instances in the security group over port 22 (SSH) for remotely logging in to Linux instances.	• Remotely log in to an instance (such as an ECS) in a security group from an external network. • Enable external servers to ping the instances in a security group to verify network connectivity. • Use instances in a security group as web servers to provide website services accessible from the Internet.
		TCP: 3389 (IPv4)	0.0.0.0/0	Allows all IPv4 addresses to access instances in a security group over port 3389 (RDP) for remotely logging in to Windows instances.	
		TCP: 80 (IPv4)	0.0.0.0/0	Allows all IPv4 addresses to access instances in a security group over port 80 (HTTP) for visiting websites.	
		TCP: 443 (IPv4)	0.0.0.0/0	Allows all IPv4 addresses to access instances in a security group over port 443 (HTTPS) for visiting websites.	

Template	Direction	Protocol/Port/Type	Source/Destination	Description	Scenario
		ICMP: All (IPv4)	0.0.0.0/0	Allows all IPv4 addresses to access instances in a security group over any port for using the ping command to test connectivity.	
		All (IPv4) All (IPv6)	Current security group	Allows the instances in a security group to communicate with each other over a private network over any protocol and port.	
	Outbound	All (IPv4) All (IPv6)	0.0.0.0/0 ::/0	Allows all traffic from the instances in the security group to external resources over any protocol and port.	
All ports open	Inbound	All (IPv4) All (IPv6)	Current security group	Allows the instances in a security group to communicate with each other over a private network over any protocol and port.	Allowing any traffic to enter and leave a security group over any port may be risky.
		All (IPv4) All (IPv6)	0.0.0.0/0 ::/0	Allows any IP address to access the instances in a security group over any protocol and port.	
	Outbound	All (IPv4) All (IPv6)	0.0.0.0/0 ::/0	Allows all traffic from the instances in the security group to external resources over any protocol and port.	

Template	Direction	Protocol/ Port/ Type	Source/ Destination	Description	Scenario
Custom rule	Inbound	All (IPv4) All (IPv6)	Current security group	Allows the instances in a security group to communicate with each other over any protocol and port.	Deny any external traffic to the instances in a security group. You can add security group rules as required.
	Outbound	All (IPv4) All (IPv6)	0.0.0.0/0 ::/0	Allows all traffic from the instances in the security group to external resources over any protocol and port.	

Procedure

1. Go to the **security group list page**.
2. In the upper right corner, click **Create Security Group**.
   The **Create Security Group** page is displayed.
3. Configure the parameters as prompted.

Figure 6-12 Create Security Group

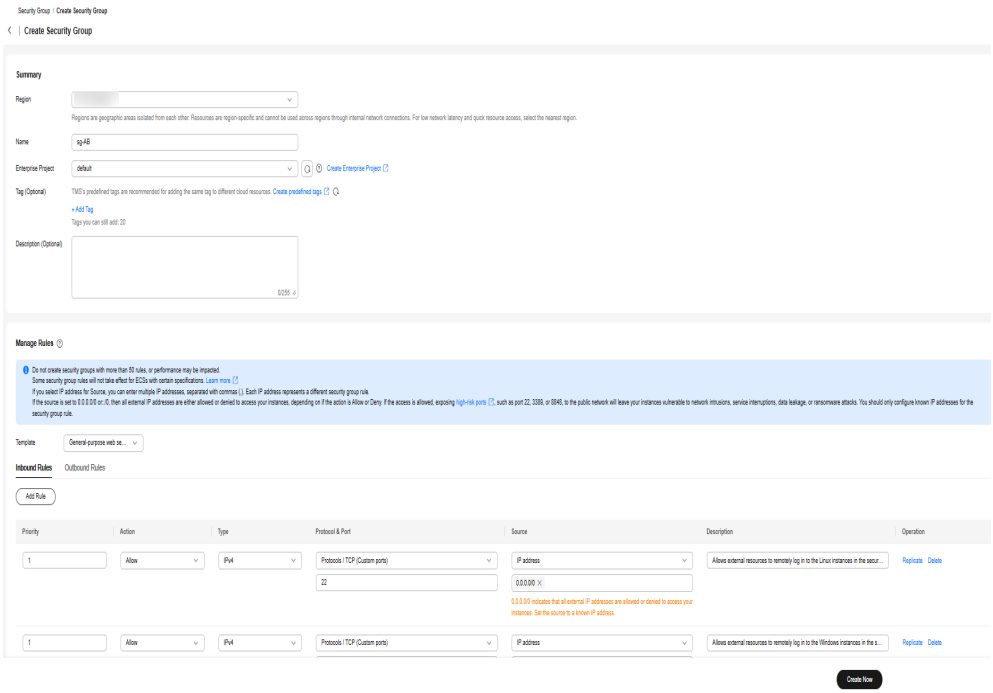


Table 6-24 Parameter description

Parameter	Description	Example Value
Region	Mandatory The region where the security group belongs. Select the region nearest to you to ensure the lowest latency possible. An instance must be in the same region as its associated security group.	CN-Hong Kong
Name	Mandatory The name of the security group. The name: • Can contain 1 to 64 characters. • Can contain letters, digits, underscores (_), hyphens (-), and periods (.). **NOTE** You can change the security group name after a security group is created. It is recommended that you give each security group a different name.	sg-AB
Enterprise Project	Mandatory When creating a security group, you can add the security group to an enabled enterprise project. An enterprise project facilitates project-level management and grouping of cloud resources and users. The name of the default project is **default**. For details about creating and managing enterprise projects, see the **Enterprise Management User Guide**.	default
Tag	Tag (Optional) You can add tags to the security group. Tags help you to identify, classify, and search for your security groups. For details, see **Managing Security Group Tags**.	**Tag key**: test **Tag value**: 01
Description (Optional)	Optional Supplementary information about the security group. The security group description can contain a maximum of 255 characters and cannot contain angle brackets (< or >).	N/A

Parameter	Description	Example Value
Template	Mandatory The preset rules have preconfigured inbound and outbound rules. You can select rules as needed. For details about preset security group rules and their application scenarios, see **Table 6-23**.	General-purpose web server

4. Check the preset rules and do the following as required:
   - Add rules.
   - Replicate rules.
   - Modify rules.
   - Delete rules.

   NOTICE

   - For details about inbound rule parameters, see **Table 6-27**. For details about outbound rule parameters, see **Table 6-28**.
   - If the source of an inbound rule is set to the current security group, you cannot delete this rule. Doing so will prevent instances in the security group from communicating with each other.
   - If the destination of an outbound rule is set to 0.0.0.0/0 or ::/0, you cannot delete this rule. Doing so will prevent instances in the security group from accessing external networks.

5. Click **Create Now**.

Follow-Up Operations

Each cloud server must be associated with at least one security group. You can add a cloud server to multiple security groups based on service requirements. For details, see **Adding an Instance to or Removing an Instance from a Security Group**.

6.2.5.2 Cloning a Security Group

Scenarios

You can clone a security group from the same or a different region to another to quickly apply the security group rules to ECSs in that region.

You can clone a security group in the following scenarios:

- For example, you have security group **sg-A** in region A. If ECSs in region B require the same security group rules as those configured for security group

sg-A, you can clone security group **sg-A** to region B, freeing you from creating a new security group in region B.

- If you need new security group rules, you can clone the original security group as a backup.
- Before you modify security group rules used by a service, you can clone the security group and modify the security group rules in the test environment to ensure that the modified rules work.

Notes and Constraints

- You can clone a security group from the same or a different region.
  - If you want to clone a security group from the same region, you can clone all rules in the security group.
  - If you want to clone a security group from a different region, the system will clone only rules whose source and destination are IP addresses and rules whose source and destination is the current security group.
- Only security group rules are cloned, but not the instances associated with the security group.
- Cloned security groups can only be used in the same account. To quickly create a security group across accounts, you can import or export security group rules by referring to **Importing and Exporting Security Group Rules**.

Procedure

1. Log in to the management console.
2. Click  in the upper left corner and select the desired region and project.
3. Click  in the upper left corner and choose **Networking** > **Virtual Private Cloud**.

   The **Virtual Private Cloud** page is displayed.
4. In the navigation pane on the left, choose **Access Control** > **Security Groups**.

   The security group list is displayed.
5. Locate the row that contains the security group, click **More** in the **Operation** column, and click **Clone**.
6. Select the region and name of the new security group as prompted.
7. Click **OK**.

   You can then switch to the required region to view the cloned security group in the security group list.

6.2.5.3 Modifying a Security Group

Scenarios

After a security group is created, you can change its name and description.

Procedure

1. Log in to the management console.

2. Click in the upper left corner and select the desired region and project.
3. Click in the upper left corner and choose **Networking** > **Virtual Private Cloud**.

   The **Virtual Private Cloud** page is displayed.
4. In the navigation pane on the left, choose **Access Control** > **Security Groups**.

   The security group list is displayed.
5. Locate the row that contains the security group, click **More** in the **Operation** column, and click **Modify**.

   The **Modify Security Group** dialog box is displayed.
6. Modify the name and description of the security group as required.
7. Click **OK** to save the modification.

6.2.5.4 Viewing the Details of a Security Group

Scenarios

You can view the details of a security group, such as the security group name, security group rules, and the instances associated with this security group.

You can also search for a given security group by key information, such as the security group name, ID, and description.

Procedure

1. Log in to the management console.
2. Click in the upper left corner and select the desired region and project.
3. Click in the upper left corner and choose **Networking** > **Virtual Private Cloud**.

   The **Virtual Private Cloud** page is displayed.
4. In the navigation pane on the left, choose **Access Control** > **Security Groups**.

   The **Security Groups** page is displayed.
5. In the search box above the security group list, select filters to quickly search for the target security group.
6. Locate the target security group and click its name.

   The security group details page is displayed.
7. On the security group details page, click different tabs to view the following information:
   - **Summary**: security group name, ID, enterprise project, and description.
   - **Inbound Rules**: the priority, action, source, and modification time of an inbound rule.
   - **Outbound Rules**: the priority, action, destination, and modification time of an outbound rule.
   - **Associated Instances**: the information about instances associated with the security group. The instances include servers, extended network interfaces, supplementary network interfaces, and others.

- **Tags**: the security group tags, including the key and value of each tag.

6.2.5.5 Managing Security Group Tags

Scenarios

Tags help you identify, classify, and search for security groups. You can perform the following operations to manage the tags of a security group:

- Add a security group tag.
- Modify a security group tag.
- Delete a security group tag.

Table 6-25 lists the details about a security group tag.

Table 6-25 Security group tag naming requirements

Parameter	Requirements	Example Value
Tag key	• For each resource, each tag key must be unique, and each tag key can only have one tag value. • Cannot be left blank. • Can contain a maximum of 128 characters. • Can contain letters in any language, digits, spaces, underscores (_), periods (.), colons (:), equal signs (=), plus signs (+), minus signs (-), and at signs (@). • Cannot start with _sys_ or a space or end with a space.	test
Tag value	• Can be left blank. • Can contain a maximum of 255 characters. • Can contain letters in any language, digits, spaces, underscores (_), periods (.), colons (:), slashes (/), equal signs (=), plus signs (+), minus signs (-), and at signs (@). • Cannot start or end with a space.	01

Notes and Constraints

Each cloud resource can have a maximum of 20 tags.

Procedure

1. Log in to the management console.
2. Click in the upper left corner and select the desired region and project.
3. Click in the upper left corner and choose **Networking** > **Virtual Private Cloud**.

   The **Virtual Private Cloud** page is displayed.
4. Go to the **security group list page**.
5. Locate the target security group and click its name.

   The security group details page is displayed.
6. On the **Tags** tab, click **Edit Tag** in the upper left corner above the tag list.

   The **Edit Tag** dialog box is displayed.
7. Perform the following operations on the tag as required:
   - Adding a tag: Click +, enter a tag key and value, and click **OK**.
   - Modifying a tag: Click × next to the target tag key or value, delete the original value, enter a new value, and click **OK**.
   - Deleting a tag: Click **Delete** next to the target tag and click **OK**.

6.2.5.6 Deleting a Security Group

Scenarios

If your security group is no longer required, you can delete it.

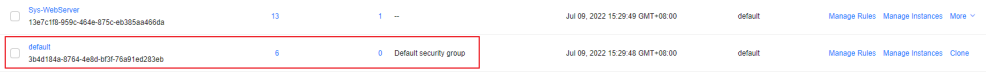
NOTE

Both default and custom security groups are free.

Notes and Constraints

- The default security group is named **default** and cannot be deleted.

  Figure 6-13 Default security group

- If you want to delete a security group that is associated with instances, such as cloud servers, containers, and databases, you need to remove the instances from the security group first. For details, see **Adding an Instance to or Removing an Instance from a Security Group**.

  If you want to know the instances associated with a security group, refer to **How Do I Know the Instances Associated with a Security Group?**
- A security group cannot be deleted if it is used as the source or destination of a rule in another security group.

  Delete or **modify** the rule and delete the security group again.

  For example, if the source of a rule in security group **sg-B** is set to **sg-A**, you need to delete or modify the rule in **sg-B** before deleting **sg-A**.

Procedure

1. Log in to the management console.
2. Click  in the upper left corner and select the desired region and project.
3. Click  in the upper left corner and choose **Networking** > **Virtual Private Cloud**.

   The **Virtual Private Cloud** page is displayed.
4. In the navigation pane on the left, choose **Access Control** > **Security Groups**.

   The security group list is displayed.
5. Locate the row that contains the target security group, click **More** in the **Operation** column, and click **Delete**.

   A confirmation dialog box is displayed.
6. Confirm the information and click **OK**.

6.2.6 Managing Security Group Rules

6.2.6.1 Adding a Security Group Rule

Scenarios

A security group consists of inbound and outbound rules. You can add security group rules to allow or deny the traffic to reach and leave the instances (such as ECSs) in the security group.

Security group rules allow or deny network traffic from specific sources over specific protocols or specific ports.

Precautions

- Before configuring security group rules, you need to plan access policies for instances in the security group. For details about common security group rules, see **Security Group Examples**.
- Add as fewer rules as possible. **Constraints** lists the constraints on the number of rules in a security group.
- After allowing traffic over a port in a security group rule, ensure that the port used by the instance is opened. For details, see **Verifying Security Group Rules**.
- If the source is set to 0.0.0.0/0 or::/0, then the access from all external IP addresses are either allowed or denied, depending on if the action is **Allow** or **Deny**. If the access is allowed, exposing **high-risk ports**, such as port 22, 3389, or 8848, to the public network will leave your instances vulnerable to network intrusions, service interruptions, data leakage, or ransomware attacks. You should only configure trusted IP addresses for the security group rule.
- By default, instances in the same security group can communicate with each other. If instances in the same security group cannot communicate with each other, possible causes are as follows:

- The inbound rules for communications between these instances are deleted. **Table 6-26** shows the inbound rules.

  Table 6-26 Inbound rules for communication between instances

Direction	Priority	Action	Type	Protocol & Port	Source/Destination
Inbound	1	Allow	IPv4	All	Source: current security group (**Sg-A**)
Inbound	1	Allow	IPv6	All	Source: current security group (**Sg-A**)

- Different VPCs cannot communicate with each other. The instances belong to the same security group but different VPCs.

  You can use **VPC peering connections** to connect VPCs in different regions.

Adding Rules to a Security Group

1. Log in to the management console.
2. Click in the upper left corner and select the desired region and project.
3. Click in the upper left corner and choose **Networking** > **Virtual Private Cloud**.

   The **Virtual Private Cloud** page is displayed.
4. In the navigation pane on the left, choose **Access Control** > **Security Groups**.

   The security group list is displayed.
5. Locate the target security group and click **Manage Rules** in the **Operation** column.

   The page for configuring security group rules is displayed.
6. On the **Inbound Rules** tab, click **Add Rule**.

   The **Add Inbound Rule** dialog box is displayed.
7. Configure required parameters.

   You can click to add more inbound rules.

Figure 6-14 Add Inbound Rule

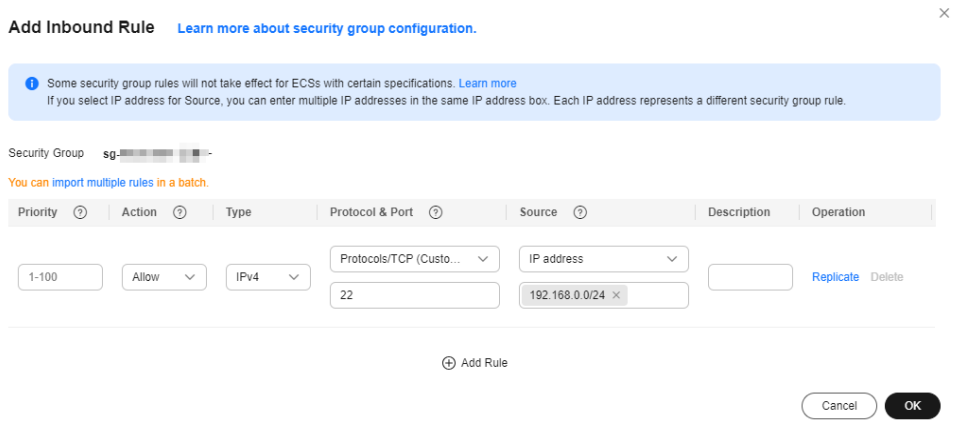


Table 6-27 Inbound rule parameter description

Parameter	Description	Example Value
Priority	The security group rule priority. The priority value ranges from 1 to 100. The default value is 1 and has the highest priority. The security group rule with a smaller value has a higher priority.	1
Action	The value can be **Allow** or **Deny**. • If the **Action** is set to **Allow**, access from the source is allowed to ECSs in the security group over specified ports. • If the **Action** is set to **Deny**, access from the source is denied to ECSs in the security group over specified ports. Security group rules are matched by priority and then by action. Deny rules take precedence over allow rules. For more information, see **How Traffic Matches Security Group Rules**.	Allow
Type	Source IP address version. You can select: • **IPv4** • **IPv6**	IPv4
Protocol & Port	The network protocol used to match traffic in a security group rule. The protocol can be **All**, **TCP**, **UDP**, **GRE**, or **ICMP**.	TCP

Parameter	Description	Example Value
	Destination port used to match traffic in a security group rule. The value can be from 1 to 65535. Inbound rules control incoming traffic over specific ports to instances in the security group. Specify one of the following: ● Individual port: Enter a port, such as **22**. ● Consecutive ports: Enter a port range, such as **22-30**. ● Non-consecutive ports: Enter ports and port ranges, such as **22,23-30**. You can enter a maximum of 20 ports and port ranges. Each port range must be unique. ● All ports: Leave it empty or enter 1-65535.	22, or 22-30

Parameter	Description	Example Value
Source	Used to match the IP address or address range of an external request. The source can be: • **IP address**: You can enter multiple IP addresses, separated by commas (,). Each IP address defines a different security group rule. – Single IP address: IP address/mask Example IPv4 address: 192.168.10.10/32 Example IPv6 address: 2002:50::44/128 – IP address range in CIDR notation: IP address/mask Example IPv4 address range: 192.168.52.0/24 Example IPv6 address range: 2407:c080:802:469::/64 – All IP addresses 0.0.0.0/0 represents all IPv4 addresses. ::/0 represents all IPv6 addresses. • **Security group**: The source is from another security group. You can select a security group in the same region from the drop-down list. If there is instance A in security group A and instance B in security group B, and the inbound rule of security group A allows traffic from security group B, traffic is allowed from instance B to instance A. • **IP address group**: An IP address group is a collection of one or more IP addresses. You can select an available IP address group from the drop-down list. An IP address group can help you manage IP address ranges and IP addresses with same security requirements in an easier way. If no IP address groups are available, create one by referring to **Creating an IP Address Group**.	IP address: 192.168.52.0/24,10.0.0.0/24
Description	Supplementary information about the security group rule. This parameter is optional. The security group rule description can contain a maximum of 255 characters and cannot contain angle brackets (< or >).	N/A

8. Click **OK**.
   The inbound rule list is displayed.
9. On the **Outbound Rules** tab, click **Add Rule**.
   The **Add Outbound Rule** dialog box is displayed.
10. Configure required parameters.
    You can click ⊕ to add more outbound rules.

Figure 6-15 Add Outbound Rule

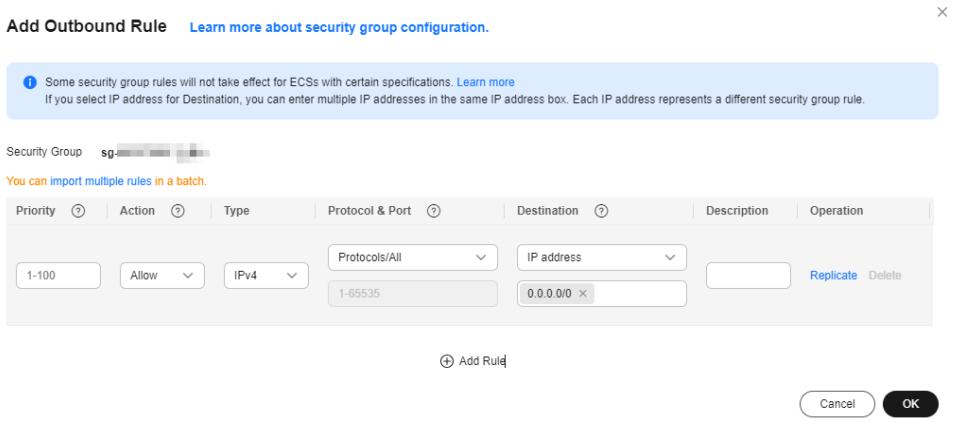


Table 6-28 Outbound rule parameter description

Parameter	Description	Example Value
Priority	The security group rule priority. The priority value ranges from 1 to 100. The default value is 1 and has the highest priority. The security group rule with a smaller value has a higher priority.	1
Action	The value can be **Allow** or **Deny**. • If the **Action** is set to **Allow**, access from ECSs in the security group is allowed to the destination over specified ports. • If the **Action** is set to **Deny**, access from ECSs in the security group is denied to the destination over specified ports. Security group rules are matched by priority and then by action. Deny rules take precedence over allow rules. For more information, see **How Traffic Matches Security Group Rules**.	Allow
Type	Destination IP address version. You can select: • **IPv4** • **IPv6**	IPv4
Protocol & Port	The network protocol used to match traffic in a security group rule. The protocol can be **All**, **TCP**, **UDP**, **GRE**, or **ICMP**.	TCP

Parameter	Description	Example Value
	Destination port used to match traffic in a security group rule. The value can be from 1 to 65535. Outbound rules control outgoing traffic over specific ports from instances in the security group. Specify one of the following: ● Individual port: Enter a port, such as **22**. ● Consecutive ports: Enter a port range, such as **22-30**. ● Non-consecutive ports: Enter ports and port ranges, such as **22,23-30**. You can enter a maximum of 20 ports and port ranges. Each port range must be unique. ● All ports: Leave it empty or enter 1-65535.	22, or 22-30

Parameter	Description	Example Value
Destination	Used to match the destination address of an internal request. The destination can be: • **IP address**: You can enter multiple IP addresses, separated by commas (,). Each IP address defines a different security group rule. – Single IP address: IP address/mask Example IPv4 address: 192.168.10.10/32 Example IPv6 address: 2002:50::44/128 – IP address range in CIDR notation: IP address/mask Example IPv4 address range: 192.168.52.0/24 Example IPv6 address range: 2407:c080:802:469::/64 – All IP addresses 0.0.0.0/0 represents all IPv4 addresses. ::/0 represents all IPv6 addresses. • **Security group**: The destination is from another security group. You can select a security group in the same region under the current account from the drop-down list. If there is instance A in security group A and instance B in security group B, and the outbound rule of security group A allows traffic to security group B, traffic is allowed from instance A to instance B. • **IP address group**: An IP address group is a collection of one or more IP addresses. You can select an available IP address group from the drop-down list. An IP address group can help you manage IP address ranges and IP addresses with same security requirements in an easier way. If no IP address groups are available, create one by referring to **Creating an IP Address Group**.	IP address: 192.168.52.0/24,10.0.0.0/24
Description	Supplementary information about the security group rule. This parameter is optional. The security group rule description can contain a maximum of 255 characters and cannot contain angle brackets (< or >).	N/A

11. Click **OK**.
    The outbound rule list is displayed.

Verifying Security Group Rules

After allowing traffic over a port in a security group rule, you need to ensure that the port used by the instance is also opened.

For example, if you have deployed a website on an ECS and want users to access your website through HTTP (80), you need to add an inbound rule to the ECS security group to allow access over the port. **Table 6-29** shows the rule.

Table 6-29 Security group rule

Direction	Priority	Action	Type	Protocol & Port	Source
Inbound	1	Allow	IPv4	TCP: 80	IP address: 0.0.0.0/0

After adding the security group rule, perform the following operations to check whether the ECS port is opened and whether the rule is applied:

1. Log in to the ECS and check whether the ECS port is opened.
   - **Checking the port of a Linux server**

     Run the following command to check whether TCP port 80 is being listened on:

     netstat -an | grep 80

     If the following figure is displayed, TCP port 80 is enabled.

     Figure 6-16 Command output for the Linux ECS

   - **Checking the port of a Windows server**

     i. Choose **Start** > **Run**. Type **cmd** to open the Command Prompt.

     ii. Run the following command to check whether TCP port 80 is being listened on:

     netstat -an | findstr 80

     If the following figure is displayed, TCP port 80 is enabled.

     Figure 6-17 Command output for the Windows ECS

2. Enter **http://***ECS EIP* in the address box of the browser and press **Enter**.

   If the requested page can be accessed, the security group rule has taken effect.

6.2.6.2 Fast-Adding Security Group Rules

Scenarios

The fast-adding rule function of security groups allows you to quickly add rules with common ports and protocols for remote login, ping tests, common web services, and database services.

Foe details about common ports used by cloud servers, see **Common ECS Ports**.

Procedure

1. Log in to the management console.
2. Click in the upper left corner and select the desired region and project.
3. Click in the upper left corner and choose **Networking** > **Virtual Private Cloud**.

   The **Virtual Private Cloud** page is displayed.
4. In the navigation pane on the left, choose **Access Control** > **Security Groups**.

   The security group list is displayed.
5. Locate the row that contains the target security group and click **Manage Rules** in the **Operation** column.

   The page for configuring security group rules is displayed.
6. On the **Inbound Rules** tab, click **Fast-Add Rule**.

   The **Fast-Add Inbound Rule** dialog box is displayed.
7. Configure required parameters.

Figure 6-18 Fast-Add Inbound Rule

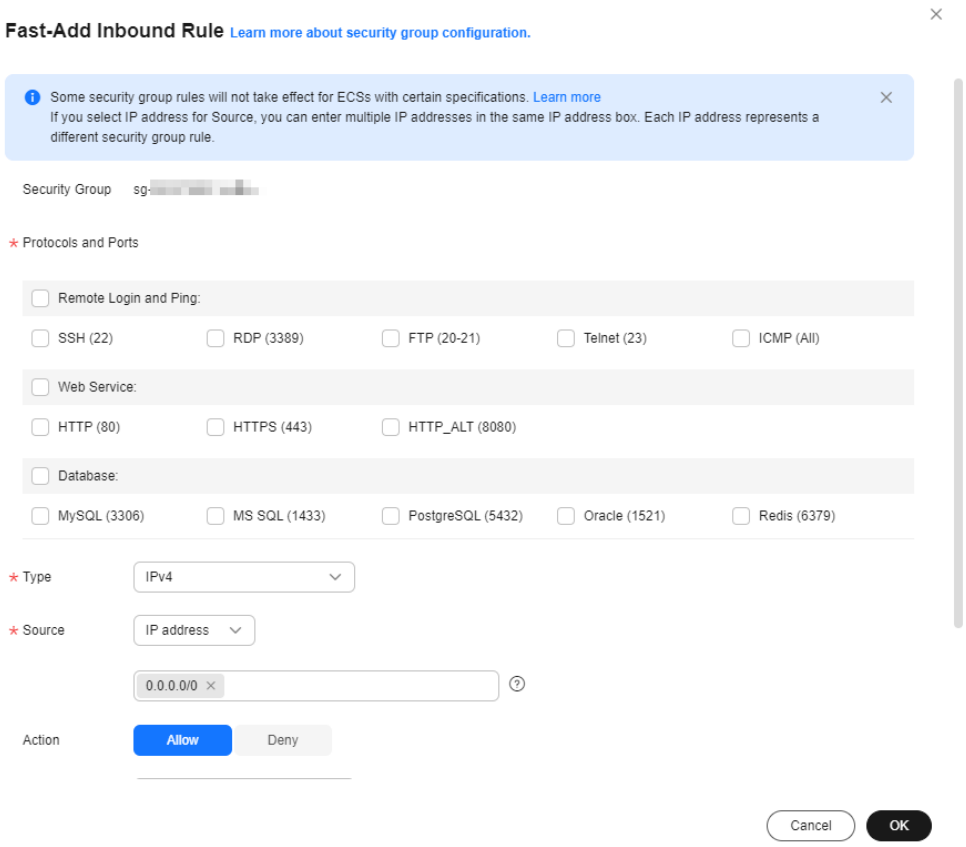


Table 6-30 Inbound rule parameter description

Parameter	Description	Example Value
Protocols and Ports	Common protocols and ports are provided for: • Remote login and ping • Web services • Databases	SSH (22)
Type	Source IP address version. You can select: • **IPv4** • **IPv6**	IPv4
Source	Used to match the IP address or address range of an external request. The source can be: • **IP address**: You can enter multiple IP addresses, separated by commas (,). Each IP address defines a different security group rule. – Single IP address: IP address/mask Example IPv4 address: 192.168.10.10/32 Example IPv6 address: 2002:50::44/128 – IP address range in CIDR notation: IP address/mask Example IPv4 address range: 192.168.52.0/24 Example IPv6 address range: 2407:c080:802:469::/64 – All IP addresses 0.0.0.0/0 represents all IPv4 addresses. ::/0 represents all IPv6 addresses. • **Security group**: The source is from another security group. You can select a security group in the same region from the drop-down list. If there is instance A in security group A and instance B in security group B, and the inbound rule of security group A allows traffic from security group B, traffic is allowed from instance B to instance A. • **IP address group**: An IP address group is a collection of one or more IP addresses. You can select an available IP address group from the drop-down list. An IP address group can help you manage IP address ranges and IP addresses with same security requirements in an easier way. If no IP address groups are available, create one by referring to **Creating an IP Address Group**.	IP address: 192.168.52.0/24,10.0.0.0/24

Parameter	Description	Example Value
Action	The value can be **Allow** or **Deny**. • If the **Action** is set to **Allow**, access from the source is allowed to ECSs in the security group over specified ports. • If the **Action** is set to **Deny**, access from the source is denied to ECSs in the security group over specified ports. Security group rules are matched by priority and then by action. Deny rules take precedence over allow rules. For more information, see **How Traffic Matches Security Group Rules**.	Allow
Priority	Security group rule priority. The priority value is from 1 to 100. The default value is 1 and has the highest priority. The security group rule with a smaller value has a higher priority.	1
Description	(Optional) Supplementary information about the security group rule. The description can contain a maximum of 255 characters and cannot contain angle brackets (< or >).	-

8. Click **OK**.
   The inbound rule list is displayed and you can view your added rule.
9. On the **Outbound Rules** tab, click **Fast-Add Rule**.
   The **Fast-Add Outbound Rule** dialog box is displayed.
10. Configure required parameters.

Figure 6-19 Fast-Add Outbound Rule

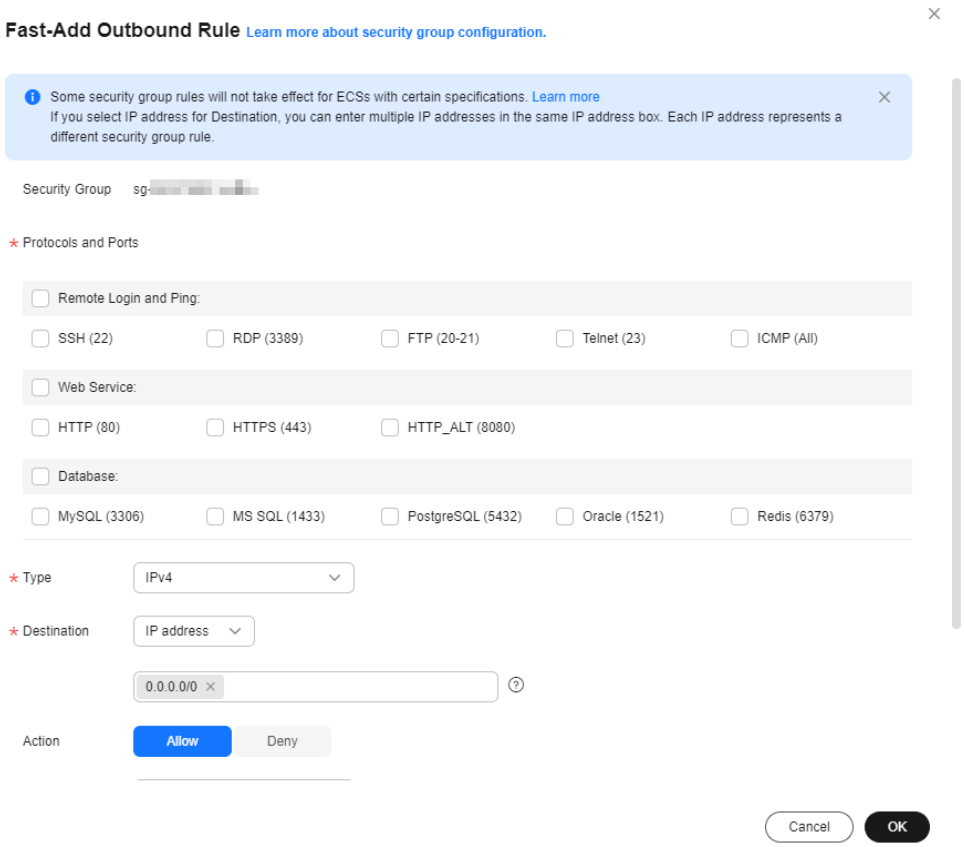


Table 6-31 Outbound rule parameter description

Parameter	Description	Example Value
Protocols and Ports	Common protocols and ports are provided for: • Remote login and ping • Web services • Databases	SSH (22)
Type	Source IP address version. You can select: • **IPv4** • **IPv6**	IPv4

Parameter	Description	Example Value
Destination	Used to match the destination address of an internal request. The destination can be: • **IP address**: You can enter multiple IP addresses, separated by commas (,). Each IP address defines a different security group rule. – Single IP address: IP address/mask Example IPv4 address: 192.168.10.10/32 Example IPv6 address: 2002:50::44/128 – IP address range in CIDR notation: IP address/mask Example IPv4 address range: 192.168.52.0/24 Example IPv6 address range: 2407:c080:802:469::/64 – All IP addresses 0.0.0.0/0 represents all IPv4 addresses. ::/0 represents all IPv6 addresses. • **Security group**: The destination is from another security group. You can select a security group in the same region under the current account from the drop-down list. If there is instance A in security group A and instance B in security group B, and the outbound rule of security group A allows traffic to security group B, traffic is allowed from instance A to instance B. • **IP address group**: An IP address group is a collection of one or more IP addresses. You can select an available IP address group from the drop-down list. An IP address group can help you manage IP address ranges and IP addresses with same security requirements in an easier way. If no IP address groups are available, create one by referring to **Creating an IP Address Group**.	IP address: 192.168.52.0/24,10.0.0.0/24
Priority	Security group rule priority. The priority value is from 1 to 100. The default value is 1 and has the highest priority. The security group rule with a smaller value has a higher priority.	1

Parameter	Description	Example Value
Action	The value can be **Allow** or **Deny**. • If the **Action** is set to **Allow**, access from ECSs in the security group is allowed to the destination over specified ports. • If the **Action** is set to **Deny**, access from ECSs in the security group is denied to the destination over specified ports. Security group rules are matched by priority and then by action. Deny rules take precedence over allow rules. For more information, see **How Traffic Matches Security Group Rules**.	Allow
Description	(Optional) Supplementary information about the security group rule. The description can contain a maximum of 255 characters and cannot contain angle brackets (< or >).	-

11. Click **OK**.
    The outbound rule list is displayed and you can view your added rule.

6.2.6.3 Allowing Common Ports with a Few Clicks

Scenarios

You can configure a security group to allow common ports with a few clicks. This function is suitable for the following scenarios:

- Remotely log in to ECSs.
- Use the ping command to test ECS connectivity.
- ECSs functioning as web servers provide website access services.

Table 6-32 describes the common ports that can be opened with a few clicks.

Table 6-32 Common ports

Direction	Protocol & Port & Type	Source/ Destination	Description
Inbound	TCP: 22 (IPv4)	0.0.0.0/0	Allows all IPv4 addresses to access ECSs in the security group over port 22 (SSH) for remotely logging in to Linux ECSs.

Direction	Protocol & Port & Type	Source/ Destination	Description
	TCP: 3389 (IPv4)	0.0.0.0/0	Allows all IPv4 addresses to access ECSs in the security group over port 3389 (RDP) for remotely logging in to Windows ECSs.
	TCP: 80 (IPv4)	0.0.0.0/0	Allows all IPv4 addresses to access ECSs in the security group over port 80 (HTTP) for visiting websites.
	TCP: 443 (IPv4)	0.0.0.0/0	Allows all IPv4 addresses to access ECSs in the security group over port 443 (HTTPS) for visiting websites.
	TCP: 20-21 (IPv4)	0.0.0.0/0	Allows all IPv4 addresses to access ECSs in the security group over ports 20 and 21 (FTP) for uploading or downloading files.
	ICMP: All (IPv4)	0.0.0.0/0	Allows all IPv4 addresses to access ECSs in the security group over any port for using the ping command to test ECS connectivity.
Outbound	All (IPv4) All (IPv6)	0.0.0.0/0 ::/0	Allows access from ECSs in the security group to any IP address over any port.

NOTICE

If the source is set to 0.0.0.0/0 or::/0, then the access from all external IP addresses are either allowed or denied, depending on if the action is **Allow** or **Deny**. If the access is allowed, exposing **high-risk ports**, such as port 22, 3389, or 8848, to the public network will leave your instances vulnerable to network intrusions, service interruptions, data leakage, or ransomware attacks. You should only configure trusted IP addresses for the security group rule.

Procedure

1. Log in to the management console.
2. Click in the upper left corner and select the desired region and project.
3. Click in the upper left corner and choose **Networking** > **Virtual Private Cloud**.

The **Virtual Private Cloud** page is displayed.

4. In the navigation pane on the left, choose **Access Control** > **Security Groups**.

   The security group list is displayed.

5. In the security group list, click the name of the security group.

   The security group details page is displayed.

6. Click the **Inbound Rules** or **Outbound Rules** tab, and then click **Allow Common Ports**.

   The **Allow Common Ports** page is displayed.

7. Click **OK**.

   After the operation is complete, you can view the added rules in the security group rule list.

6.2.6.4 Modifying a Security Group Rule

Scenarios

You can modify the port, protocol, and IP address of your security group rules as required to ensure the security of your instances.

Note that modifying a security group rule may interrupt your services or cause network security risks.

Notes and Constraints

Security group rules are like a whitelist. If there are no rules that allow or deny specific traffic, the security group denies all traffic to or from the instances in it.

- The inbound rules in **Table 6-33** ensure that instances in the security group can communicate with each other. Do not modify these rules.
- The outbound rules in **Table 6-33** allow instances in the security group to access external networks. If you modify these rules, the instances in the security group cannot access external networks.

Table 6-33 Security group rules

Direction	Action	Type	Protocol & Port	Source/Destination
Inbound	Allow	IPv4	All	Source: current security group
Inbound	Allow	IPv6	All	Source: current security group
Outbound	Allow	IPv4	All	Destination: 0.0.0.0/0
Outbound	Allow	IPv6	All	Destination: ::/0

Procedure

1. Log in to the management console.
2. Click in the upper left corner and select the desired region and project.
3. Click in the upper left corner and choose **Networking** > **Virtual Private Cloud**.
   The **Virtual Private Cloud** page is displayed.
4. In the navigation pane on the left, choose **Access Control** > **Security Groups**.
   The security group list is displayed.
5. In the security group list, click the name of the security group.
   The security group details page is displayed.
6. Click the **Inbound Rules** or **Outbound Rules** tab as required.
   The security group rule list is displayed.
7. Locate the target rule and click **Modify** in the **Operation** column.
8. Modify the security group rule information as prompted and click **Confirm**.

6.2.6.5 Replicating a Security Group Rule

Scenarios

You can replicate an existing security group rule and modify it to quickly generate a new rule.

Procedure

1. Log in to the management console.
2. Click in the upper left corner and select the desired region and project.
3. Click in the upper left corner and choose **Networking** > **Virtual Private Cloud**.
   The **Virtual Private Cloud** page is displayed.
4. In the security group list, click the name of the security group.
   The security group details page is displayed.
5. Click the **Inbound Rules** or **Outbound Rules** tab as required.
   The security group rule list is displayed.
6. Locate the target rule and click **Replicate** in the **Operation** column.
   The **Replicate Inbound Rule** or **Replicate Outbound Rule** dialog box is displayed.
7. Modify the security group rule information as prompted and click **OK**.

6.2.6.6 Enabling or Disabling One or More Security Group Rules

Scenarios

After a security group rule is added, it is enabled by default. You can enable or disable a security group rule as needed.

- After a security group rule is disabled, it will not work. If all security group rules are disabled, traffic will be denied to reach or leave the instances in the security group. Disabling all rules may interrupt network traffic.
- After a security group rule is enabled, it controls the network traffic from and to the instances in the security group.

To enable or disable a security group rule, see **Enabling or Disabling a Security Group Rule**.

To enable or disable multiple security group rules at a time:

- If there are a small number of rules to be enabled or disabled, you can select these rules in the security group rule list on the console by referring to **Enabling or Disabling Multiple Security Group Rules Directly on the Console**.
- If there are a large number of rules to be enabled or disabled, you can **export the rule list to a local Excel file**, only keep the rules you want to enable or disable, and import the file to the console. The system then selects the rules to be processed based on the imported file. For details, see **Enabling or Disabling Multiple Security Group Rules Using an Excel File**.

Enabling or Disabling a Security Group Rule

1. Log in to the management console.
2. Click  in the upper left corner and select the desired region and project.
3. Click  in the upper left corner and choose **Networking** > **Virtual Private Cloud**.

   The **Virtual Private Cloud** page is displayed.
4. In the navigation pane on the left, choose **Access Control** > **Security Groups**.

   The security group list is displayed.
5. Locate the target security group and click its name.

   The security group details page is displayed.
6. Click the **Inbound Rules** or **Outbound Rules** tab as required.

   The security group rule list is displayed.
7. In the security group rule list:
   - Enable a security group rule.
     i. Locate the target security group rule, click **More** in the **Operation** column, and select **Enable**.

        A confirmation dialog box is displayed.

     ii. Click **OK**.
   - Disable a security group rule.
     i. Locate the target security group rule, click **More** in the **Operation** column, and select **Disable**.

        A confirmation dialog box is displayed.

     ii. Click **OK**.

Enabling or Disabling Multiple Security Group Rules Directly on the Console

1. Log in to the management console.
2. Click  in the upper left corner and select the desired region and project.
3. Click in the upper left corner and choose **Networking** > **Virtual Private Cloud**.

   The **Virtual Private Cloud** page is displayed.
4. In the navigation pane on the left, choose **Access Control** > **Security Groups**.

   The security group list is displayed.
5. Locate the target security group and click its name.

   The security group details page is displayed.
6. Click the **Inbound Rules** or **Outbound Rules** tab as required.

   The security group rule list is displayed.
7. In the security group rule list, select the security group rules you want to enable or disable.
8. In the rule list:
   - Enable security group rules.
     i. Above the security group rule list, choose **More** > **Enable**.

        A confirmation dialog box is displayed.

     ii. Click **OK**.
   - Disable security group rules.
     i. Above the security group rule list, choose **More** > **Disable**.

        A confirmation dialog box is displayed.

     ii. Click **OK**.

Enabling or Disabling Multiple Security Group Rules Using an Excel File

1. Log in to the management console.
2. Click  in the upper left corner and select the desired region and project.
3. Click in the upper left corner and choose **Networking** > **Virtual Private Cloud**.

   The **Virtual Private Cloud** page is displayed.
4. In the navigation pane on the left, choose **Access Control** > **Security Groups**.

   The security group list is displayed.
5. Locate the target security group and click its name.

   The security group details page is displayed.
6. Click the **Inbound Rules** or **Outbound Rules** tab as required.

   The security group rule list is displayed.
7. In the upper left corner above the security group rule list, click **Batch Operations**.

   The **Batch Operations** dialog box is displayed.

8. Select either of the following methods:
   - Method 1: Click **Download Template** to download the Excel file to your local PC and fill in the security group rules to be enabled or disabled in the file.
   - Method 2: **Export the existing rules to a local Excel file**, filter the target rules and keep them as they are, and save the file.

   After the Excel file is ready, take step **9**. The system then automatically selects the target rules based on the imported file.
9. In the **Batch Operations** dialog box, click **Select File**.

   The system starts to match the rules in the Excel file against existing security group rules based on the priority, action, type, protocol & port, source, and destination.
   - If a rule in the Excel file matches an existing rule, **Verified** is displayed in the **Result** column. Only the matched rules can be enabled or disabled.
   - If a rule fails to be matched, the causes will be displayed in the **Result** column. The possible causes are as follows:
     - There is no such rule in this security group.
     - Inconsistent rule direction. For example, you perform the operation on outbound rules on the **Inbound Rules** tab, or the other way around.
     - Duplicate rules in the Excel file. The system automatically filters out the duplicate rules.
10. Confirm the rules and click **OK**.

    The security group rule list page is displayed and the target rules are selected automatically.
11. In the rule list:
    - Enable security group rules.
      i. Above the security group rule list, choose **More** > **Enable**.

      A confirmation dialog box is displayed.

      ii. Click **OK**.
    - Disable security group rules.
      i. Above the security group rule list, choose **More** > **Disable**.

      A confirmation dialog box is displayed.

      ii. Click **OK**.

6.2.6.7 Importing and Exporting Security Group Rules

Scenarios

You can configure security group rules in an Excel file and import the rules to a security group. You can also export security group rules to an Excel file.

You can import and export security group rules in the following scenarios:

- If you want to back up security group rules locally, you can export the rules to an Excel file.

- If you want to quickly create or restore security group rules, you can import your security group rule file to the security group.
- If you want to quickly apply the rules of one security group to another, you can export and import existing rules.
- If you want to modify multiple rules of the current security group at a time, you can export and import existing rules.

Notes and Constraints

- The security group rules to be imported must be configured based on the template. Do not add parameters or change existing parameters. Otherwise, the import will fail.
- If you import a security group rule with **Source**/**Destination** set to a security group or IP address group, ensure that the group ID is correct. Otherwise, the import will fail.
- Duplicate security group rules will be ignored during import, whether they already exist in the security group or are included in the rules to be imported. As described in **Table 6-34**, rules A, B, and C are duplicate rules.
  - Rules A and B have the same direction, action, type, protocol & port, source address, and destination address but different priorities.
  - Rules A and C have the same direction, priority, action, type, protocol & port, source address, and destination address.

  Table 6-34 Duplicate rules

Rule	Direction	Priority	Action	Type	Protocol & Port	Destination
Rule A	Inbound	1	Allow	IPv4	TCP: 22	0.0.0.0/0
Rule B	Inbound	5	Allow	IPv4	TCP: 22	0.0.0.0/0
Rule C	Inbound	1	Allow	IPv4	TCP: 22	0.0.0.0/0

- Do not import two security group rules with the same **Direction**, **Type**, **Protocol & Port**, and **Source**/**Destination**, but different **Action** configurations. **Table 6-35** shows an example.
  - If a rule to be imported conflicts with an existing rule in the security group, the import will fail. In this case, rectify the fault as prompted.
  - If rules to be imported conflicts with each other, the import will fail. In this case, rectify the fault as prompted.

Table 6-35 Rules with different actions

Rule	Direction	Priority	Action	Type	Protocol & Port	Destination
Rule A	Inbound	1	Allow	IPv4	TCP: 22	0.0.0.0/0
Rule B	Inbound	5	Deny	IPv4	TCP: 22	0.0.0.0/0

- If you want to import rules of the security group in one region to another under one account, only rules with both **Source** and **Destination** set to **IP address** can be applied.
- If you want to import rules of the security group in one account to the security group in another account, only rules with both **Source** and **Destination** set to **IP address** can be applied.

Procedure

1. Log in to the management console.
2. Click  in the upper left corner and select the desired region and project.
3. Click  in the upper left corner and choose **Networking** > **Virtual Private Cloud**.

   The **Virtual Private Cloud** page is displayed.
4. In the navigation pane on the left, choose **Access Control** > **Security Groups**.

   The security group list is displayed.
5. On the security group list, click the name of the target security group.

   The security group details page is displayed.
6. Export and import security group rules.
   - Click **Export Rule** to export all rules of the current security group to an Excel file.
   - Click **Import Rule** to import security group rules from an Excel file into the current security group.

     Table 6-36 describes the parameters in the template for importing rules.

     Table 6-36 Template parameters

Parameter	Description	Example Value
Direction	The direction in which the security group rule takes effect. • **Inbound**: Inbound rules control incoming traffic to instances in the security group. • **Outbound**: Outbound rules control outgoing traffic from instances in the security group.	Inbound

Parameter	Description	Example Value
Priority	The priority value ranges from 1 to 100. The default value is 1 and has the highest priority. The security group rule with a smaller value has a higher priority.	1
Action	The value can be **Allow** or **Deny**. • If the **Action** is set to **Allow**, access from the source is allowed to ECSs in the security group over specified ports. • If the **Action** is set to **Deny**, access from the source is denied to ECSs in the security group over specified ports. Security group rules are matched by priority and then by action. Deny rules take precedence over allow rules. For more information, see **How Traffic Matches Security Group Rules**.	Allow
Protocol & Port	The network protocol used to match traffic in a security group rule. The protocol can be **All**, **TCP**, **UDP**, **GRE**, or **ICMP**.	TCP
	Destination port used to match traffic in a security group rule. The value can be from 1 to 65535. Inbound rules control incoming traffic over specific ports to instances in the security group. Outbound rules control outgoing traffic over specific ports from instances in the security group.	22, or 22-30
Type	Source IP address version. You can select: • **IPv4** • **IPv6**	IPv4

Parameter	Description	Example Value
Source	The source in an inbound rule is used to match the IP address or address range of an external request. The source can be: • **IP address**: – Single IP address: IP address/mask Example IPv4 address: 192.168.10.10/32 Example IPv6 address: 2002:50::44/128 – IP address range in CIDR notation: IP address/mask Example IPv4 address range: 192.168.52.0/24 Example IPv6 address range: 2407:c080:802:469::/64 – All IP addresses 0.0.0.0/0 represents all IPv4 addresses. ::/0 represents all IPv6 addresses. • **Security group**: The source is from another security group. You can select a security group in the same region under the current account. Instance A is in security group A and instance B is in security group B. If security group A has an inbound rule with **Action** set to **Allow** and **Source** set to security group B, access from instance B is allowed to instance A. A security group is in the format of *Security group name(Security group ID)*. An example is sg-test(96a8a93f-XXX-d7872990c314). • **IP address group**: An IP address group is a collection of one or more IP addresses. You can select an available IP address group. An IP address group can help you manage IP address ranges and IP addresses with same security requirements in an easier way. A security group is in the format of *IP address group name(IP address group ID)*. An example is ipGroup-test(96a8a93f-XXX-d7872990c314).	sg-test[96a8a93f-XXX-d7872990c314]

Parameter	Description	Example Value
Destination	The destination in an outbound rule is used to match the IP address or address range of an internal request. The destination can be: • **IP address** – Single IP address: IP address/mask Example IPv4 address: 192.168.10.10/32 Example IPv6 address: 2002:50::44/128 – IP address range in CIDR notation: IP address/mask Example IPv4 address range: 192.168.52.0/24 Example IPv6 address range: 2407:c080:802:469::/64 – All IP addresses 0.0.0.0/0 represents all IPv4 addresses. ::/0 represents all IPv6 addresses. • **Security group**: The destination is from another security group. Instance A is in security group A and instance B is in security group B. If security group A has an outbound rule with **Action** set to **Allow** and **Destination** set to security group B, access from instance A is allowed to instance B. A security group is in the format of *Security group name(Security group ID)*. An example is sg-test(96a8a93f-XXX-d7872990c314). • **IP address group**: An IP address group is a collection of one or more IP addresses. An IP address group can help you manage IP address ranges and IP addresses with same security requirements in an easier way. A security group is in the format of *IP address group name(IP address group ID)*. An example is ipGroup-test(96a8a93f-XXX-d7872990c314).	sg-test[96a8a93f-XXX-d7872990c314]
Description	Supplementary information about the security group rule. This parameter is optional. The security group rule description can contain a maximum of 255 characters and cannot contain angle brackets (< or >).	-
Last Modified	The time when the security group was modified.	-

6.2.6.8 Deleting One or More Security Group Rules

Scenarios

If you no longer need one or more security group rules to control the traffic to and from the instances in a security group, you can delete them.

To delete a security group rule, see **Deleting a Security Group Rule**.

To delete multiple security group rules at a time:

- If there are a small number of rules to be deleted, you can select these rules in the rule list on the console by referring to **Deleting Multiple Security Group Rules Directly on the Console**.
- If there are a large number of rules to be deleted, you can **export the rule list to a local Excel file**, only keep the rules you want to delete, and import the file to the console. The system then selects the rules to be deleted based on the imported file. For details, see **Deleting Multiple Security Group Rules Using an Excel File**.

Notes and Constraints

Note that deleting a security group rule may interrupt your services or cause network security risks.

Security group rules are like a whitelist. If there are no rules that allow or deny specific traffic, the security group denies all traffic to or from the instances in it.

- The inbound rules in **Table 6-37** ensure that instances in the security group can communicate with each other. Do not delete these rules.
- The outbound rules in **Table 6-37** allow instances in the security group to access external networks. If you delete these rules, the instances in the security group cannot access external networks.

Table 6-37 Security group rules

Direction	Action	Type	Protocol & Port	Source/Destination
Inbound	Allow	IPv4	All	Source: current security group
Inbound	Allow	IPv6	All	Source: current security group
Outbound	Allow	IPv4	All	Destination: 0.0.0.0/0
Outbound	Allow	IPv6	All	Destination: ::/0

Deleting a Security Group Rule

1. Log in to the management console.
2. Click  in the upper left corner and select the desired region and project.

3. Click ☰ in the upper left corner and choose **Networking** > **Virtual Private Cloud**.
   The **Virtual Private Cloud** page is displayed.
4. In the navigation pane on the left, choose **Access Control** > **Security Groups**.
   The security group list is displayed.
5. In the security group list, click the name of the security group.
   The security group details page is displayed.
6. Click the **Inbound Rules** or **Outbound Rules** tab as required.
   The security group rule list is displayed.
7. Locate the target rule and click **Delete** in the **Operation** column.
   A confirmation dialog box is displayed.
8. Click **OK**.

Deleting Multiple Security Group Rules Directly on the Console

1. Log in to the management console.
2. Click in the upper left corner and select the desired region and project.
3. Click ☰ in the upper left corner and choose **Networking** > **Virtual Private Cloud**.
   The **Virtual Private Cloud** page is displayed.
4. In the navigation pane on the left, choose **Access Control** > **Security Groups**.
   The security group list is displayed.
5. In the security group list, click the name of the security group.
   The security group details page is displayed.
6. Click the **Inbound Rules** or **Outbound Rules** tab as required.
   The security group rule list is displayed.
7. In the security group rule list, select the target security group rules and click **Delete** up above the upper left corner of the list.
   A confirmation dialog box is displayed.
8. Click **OK**.

Deleting Multiple Security Group Rules Using an Excel File

1. Log in to the management console.
2. Click in the upper left corner and select the desired region and project.
3. Click ☰ in the upper left corner and choose **Networking** > **Virtual Private Cloud**.
   The **Virtual Private Cloud** page is displayed.
4. In the navigation pane on the left, choose **Access Control** > **Security Groups**.
   The security group list is displayed.
5. In the security group list, click the name of the security group.
   The security group details page is displayed.

6. Click the **Inbound Rules** or **Outbound Rules** tab as required.

   The security group rule list is displayed.

7. In the upper left corner above the security group rule list, click **Batch Operations**.

   The **Batch Operations** dialog box is displayed.

8. Select either of the following methods:
   - Method 1: Click **Download Template** to download the Excel file to your local PC and fill in the security group rules to be enabled or disabled in the file.
   - Method 2: **Export the existing rules to a local Excel file**, filter the target rules and keep them as they are, and save the file.

   After the Excel file is ready, take step **9**. The system then automatically selects the target rules based on the imported file.

9. In the **Batch Operations** dialog box, click **Select File**.

   The system starts to match the rules in the Excel file against existing security group rules based on the priority, action, type, protocol & port, source, and destination.
   - If a rule in the Excel file matches an existing rule, **Verified** is displayed in the **Result** column. Only the matched rules can be enabled or disabled.
   - If a rule fails to be matched, the causes will be displayed in the **Result** column. The possible causes are as follows:
     - There is no such rule in this security group.
     - Inconsistent rule direction. For example, you perform the operation on outbound rules on the **Inbound Rules** tab, or the other way around.
     - Duplicate rules in the Excel file. The system automatically filters out the duplicate rules.

10. Confirm the rules and click **OK**.

    The security group rule list page is displayed and the target rules are selected automatically.

11. In the upper left corner above the security group rule list, click **Delete**.

    A confirmation dialog box is displayed.

12. Click **OK**.

6.2.6.9 Querying Security Group Rule Changes

Scenarios

CTS records the changes made to security group rules. You can query the change details of:

- New security group rules
- Modified security group rules
- Deleted security group rules

Precautions

- To use CTS to record security group rule changes, you need to **enable CTS** first.
- CTS records operations performed on each cloud service. You can query specific operations by trace name, resource type, or operation time. **Table 6-38** lists the operations on security group rules supported by CTS.

Table 6-38 Operations on security group rules supported by CTS

Operation	Trace Name	Resource Type
Adding a security group rule	createSecurity-group-rule	security-group-rules
Modifying a security group rule	updateSecurity-group-rule	security-group-rules
Deleting a security group rule	deleteSecurity-group-rule	security-group-rules

Procedure

The following describes how to view the rule described in **Table 6-39** that is added to security group **Sg-A**.

Table 6-39 The new security group rule

Direction	Action	Type	Protocol & Port	Source	Last Modified
Inbound	Allow	IPv4	TCP: 23	10.0.0.0/16	June 19, 2024 10:46:07 GMT+08:00

1. Log in to the CTS console, search for the operations by trace name (**createSecurity-group-rule** in this example) and locate the specific operation by operation time.

   For details, see **Querying Real-Time Traces**.

Figure 6-20 The trace list for new security group rules

2. In the trace list, locate the target trace and click its name.

   On the **Trace Overview** page, you can view the details about the operation. **Table 6-40** provides the detailed information about the operation, including operator ID and details about the security group rules.

   NOTE

   The trace details in **Table 6-40** are only for your reference. The actual information may vary.

Table 6-40 The trace details for the new security group rule

Example Command Output	Description
"source_ip": "124.71.XX.146",	IP address of the client that performs the operation. If this parameter is left blank, the operation is performed by the system. In this example, the IP address is **124.71.XX.146**.
"user": { "access_key_id": "HSTA205XXXXXC4MHAE", "account_id": "3c24f6f885294XXXXX93ce075fbd", "user_name": "cts-test-01", **"domain": {** **"name": "cts-test",** **"id": "3c24f6f885294XXXXX93ce075fbd"** **},** **"name": "cts-test-01",** "principal_is_root_user": "false", **"id": "a26ee7e7224XXXXXe4a28a9ce503",**	Account of the operator who performs the operation. Key parameters are described as follows: • **name** under **domain**: indicates the account name. In this example, the account name is **cts-test**. • **id** under **domain**: indicates the account ID. In this example, the ID is **3c24f6f885294XXXXX93ce075fbd**. • **name**: IAM username. In this example, the username is **cts-test-01**, which is an IAM user under account **cts-test**. • **id**: IAM user ID. In this example, the ID is **a26ee7e7224XXXXXe4a28a9ce503**. For details about more parameters of CTS traces, see the response parameter description in **Trace Structure**.

Example Command Output	Description
"response": "{\"request_id\":\"8d2d1111cafaXXX9b49d53e2da38f\",\"security_group_rules\":[{\"id\":\"b6acda6e-0976-XXXX-82bc-a8093cbd591d\",\"project_id\":\"15289aca74eXXXa37dea0315d99\",\"security_group_id\":\"3730d371-3111-4ace-XXXX-b00b7259e178\",\"remote_group_id\":null,**\"direction\":\"ingress\",\"protocol\":\"tcp\"**,\"description\":\"\",\"created_at\":\"2024-06-19T02:46:07Z\",\"updated_at\":\"2024-06-19T02:46:07Z\",**\"ethertype\":\"IPv4\",\"remote_ip_prefix\":\"10.0.0.0/16\",\"multiport\":\"23\"**,\"remote_address_group_id\":null,**\"action\":\"allow\",\"priority\":1**}]}",	Details about the security group rule in **response**. Key parameters are described as follows: • **direction**: indicates the direction of the security group rule. **ingress** indicates the inbound direction, and **egress** indicates the outbound direction. In this example, **ingress** is returned, indicating an inbound rule is added. • **protocol**: indicates the protocol of the security group rule. In this example, the protocol is **TCP**. • **ethertype**: indicates the source IP address version. In this example, the version is **IPv4**. • **remote_ip_prefix**: indicates the source or destination of the security group rule. In this example, an inbound rule is added, so this parameter indicates IP address range **10.0.0.0/16**. • **multiport**: indicates the port used to filter traffic. In this example, the port is **23**. • **action**: indicates whether to allow or deny traffic. **allow** indicates traffic is allowed, while **deny** indicates traffic is denied. In this example, the action is **allow**. • **priority**: indicates the priority of the security group rule. In this example, the priority is **1**. For details about more parameters of security group rules, see **Querying a Security Group Rule**.

6.2.7 Managing Instances Added to a Security Group

6.2.7.1 Adding an Instance to or Removing an Instance from a Security Group

Scenarios

When you create an instance, the system automatically adds the instance to a security group for protection.

- If one security group cannot meet your requirements, you can add an instance to multiple security groups.
- An instance must be added to at least one security group. If you want to change the security group for an instance, you can add the instance to a new security group and then remove the instance from the original security group.

You can add servers, extended network interfaces, and supplementary network interfaces to a security group by referring to the following operations:

- **Adding an Instance to a Security Group**
- **Removing an Instance from a Security Group**

Notes and Constraints

If you see a message saying you lack the required permissions when viewing a security group's resources on the management console, you need to request the permissions for viewing the security group and its associated resources, such as servers, extended network interfaces, and supplementary network interfaces. For details, see **Example 4: Allowing users to view associated resources**.

Adding an Instance to a Security Group

1. Log in to the management console.
2. Click  in the upper left corner and select the desired region and project.
3. Click  in the upper left corner and choose **Networking** > **Virtual Private Cloud**.

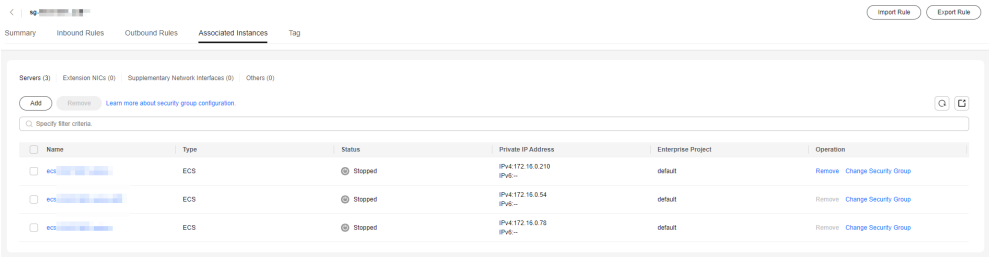
   The **Virtual Private Cloud** page is displayed.
4. In the navigation pane on the left, choose **Access Control** > **Security Groups**.

   The security group list is displayed.
5. In the security group list, locate the row that contains the security group and click **Manage Instances** in the **Operation** column.

   The **Associated Instances** tab is displayed.
6. Click the required instance type tab.

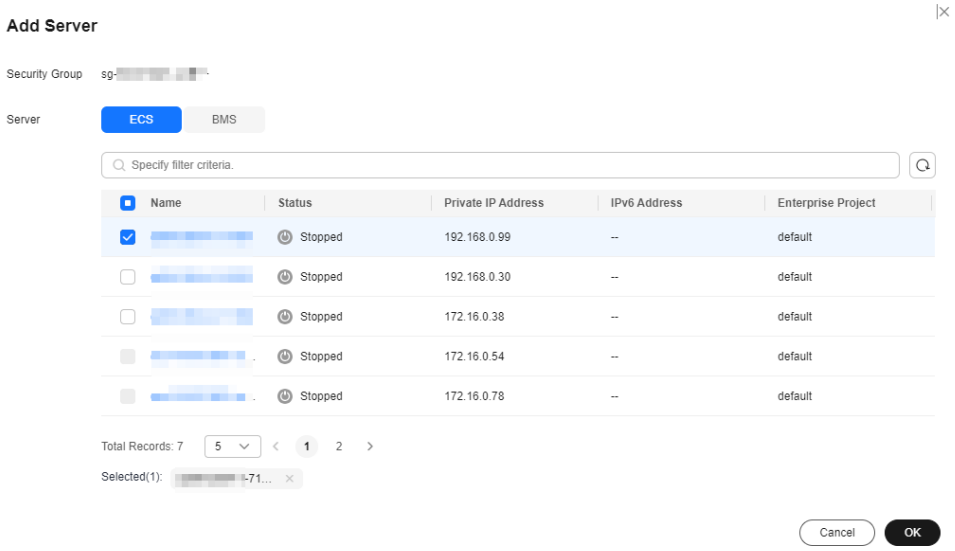
   The following operations use **Servers** as an example.

Figure 6-21 Associated Instances (Servers)



7. Click the **Servers** tab and click **Add**.
   The **Add Server** dialog box is displayed.

Figure 6-22 Adding cloud servers



8. In the server list, select one or more servers and click **OK** to add them to the current security group.

Removing an Instance from a Security Group

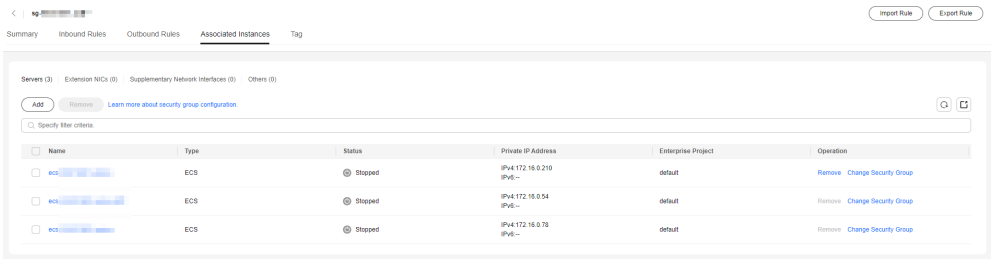
An instance must be added to at least one security group. If you want to remove an instance from a security group, the instance must be associated with at least two security groups now.

1. Log in to the management console.
2. Click in the upper left corner and select the desired region and project.
3. Click in the upper left corner and choose **Networking** > **Virtual Private Cloud**.
   The **Virtual Private Cloud** page is displayed.
4. In the navigation pane on the left, choose **Access Control** > **Security Groups**.
   The security group list is displayed.
5. In the security group list, locate the row that contains the security group and click **Manage Instances** in the **Operation** column.
   The **Associated Instances** tab is displayed.

6. Click the required instance type tab.

   The following operations use **Servers** as an example.

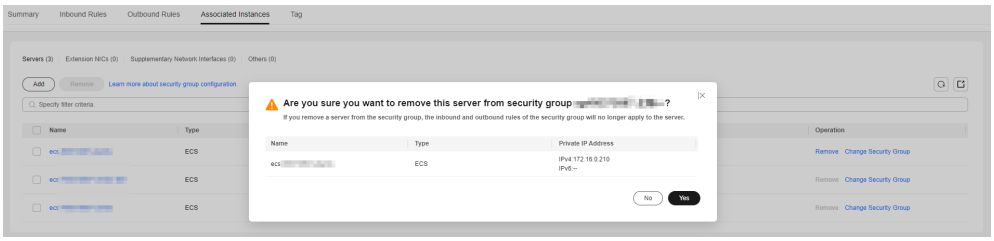
   Figure 6-23 Associated Instances (Servers)



7. Click the **Servers** tab, select one or more servers, and click **Remove** in the upper left corner of the server list.

   A confirmation dialog box is displayed.

   Figure 6-24 Removing cloud servers



8. Confirm the information and click **OK**.

6.2.7.2 Changing the Security Group of an ECS

Scenarios

When creating an ECS, you must associate it with a security group. If no security group has been created yet, a **default security group** will be created and associated with the ECS. If the default security group cannot meet your service requirements, you can change the security group associated with the ECS.

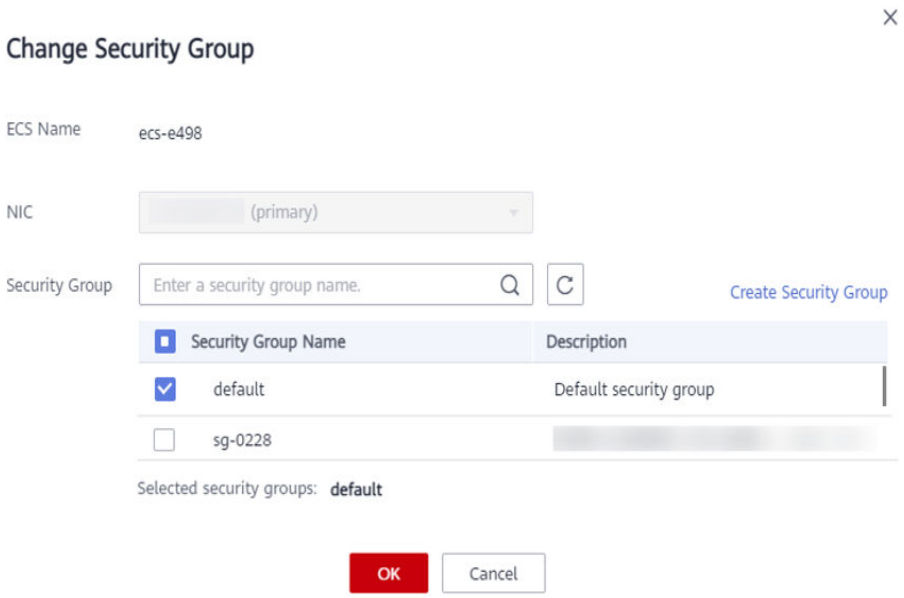
You can also associate an ECS with a custom security group. If the custom security group cannot meet your requirements, you can change the custom security group.

Procedure

1. Log in to the management console.
2. Click . Under **Compute**, click **Elastic Cloud Server**.
3. In the ECS list, locate the row that contains the target ECS. Click **More** in the **Operation** column and select **Manage Network** > **Change Security Group**.

   The **Change Security Group** dialog box is displayed.

Figure 6-25 Change Security Group



4. Select the target NIC and security groups.

   You can select multiple security groups. In such a case, the rules of all the selected security groups will apply to the ECS.

   To create a security group, click **Create Security Group**.

   NOTE

   Using multiple security groups may deteriorate ECS network performance. You are suggested to select no more than five security groups.

5. Click **OK**.

6.3 Network ACL

6.3.1 Network ACL Overview

Network ACL

A network ACL is an optional layer of protection for your subnets. After you add inbound and outbound rules to a network ACL and associate subnets with it, you can control traffic in and out of the subnets.

A network ACL is different from a security group. A security group protects the instances in it, such as ECSs, databases, and containers, while a network ACL protects the entire subnet. Security groups are a mandatory layer of protection but network ACLs are optional. Network ACLs and security groups can be used together for fine-grained access control.

You need to specify the protocol, source port and address, and destination port and address for each inbound and outbound rule of the network ACL. Suppose you have two subnets in region A, as shown in **Figure 6-26**. **Subnet-X01** is associated with network ACL **Fw-A**, and ECSs deployed in this subnet provide web

services accessible from the Internet. **Subnet-X02** is associated with network ACL **Fw-B**. **Subnet-X02** and **Subnet-Y01** are connected through a VPC peering connection. Now, you need to configure inbound and outbound rules to allow **ECS-C01** in **Subnet-Y01** to remotely log in to ECSs in **Subnet-X02**.

- Inbound and outbound rules on **Fw-A**:

  Custom inbound rule **A01** allows any IP address to access the ECSs in **Subnet-X01** through port 80 over TCP or HTTP. If the traffic does not match custom rule **A01**, the default rule is applied and the traffic is denied to flow into the subnet.

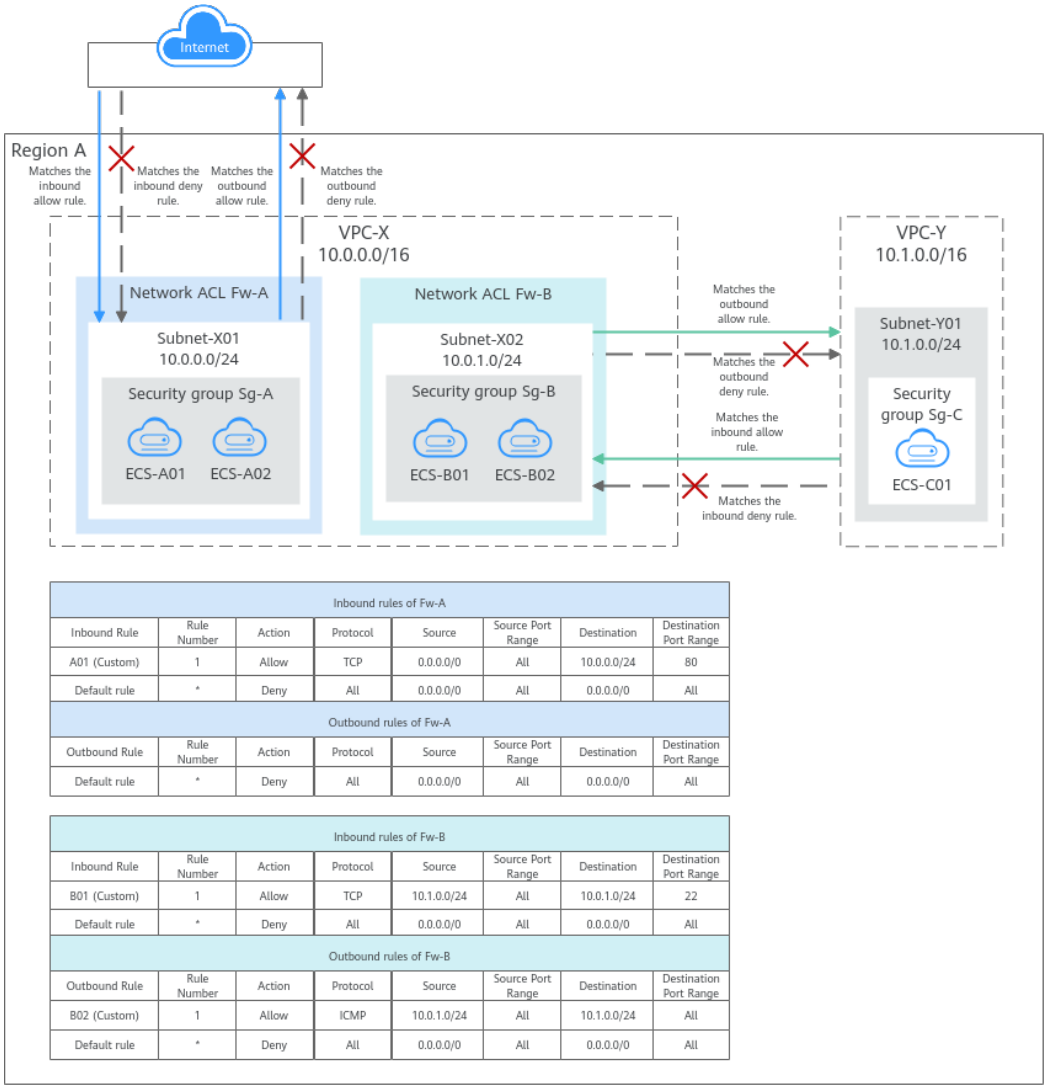
  Stateful network ACLs allow responses to inbound requests to leave the subnet without being controlled by rules. The responses from ECSs in **Subnet-X01** can go out of the subnet. Other outbound traffic is not allowed to leave **Subnet-X01**, because the default rule is applied.

- Inbound and outbound rules on **Fw-B**:

  Custom inbound rule **B01** allows **ECS-C01** in **Subnet-Y01** to use access the ECSs in **Subnet-X02** through port 22 over TCP or SSH.

  Custom outbound rule **B02** allows all ICMP traffic over any port. The ping traffic from ECSs in **Subnet-X02** to **ECS-C01** in **Subnet-Y01** can be routed successfully to test the network connectivity.

Figure 6-26 Network ACL rules



Inbound rules of Fw-A							
Inbound Rule	Rule Number	Action	Protocol	Source	Source Port Range	Destination	Destination Port Range
A01 (Custom)	1	Allow	TCP	0.0.0.0/0	All	10.0.0.0/24	80
Default rule	*	Deny	All	0.0.0.0/0	All	0.0.0.0/0	All
Outbound rules of Fw-A							
Outbound Rule	Rule Number	Action	Protocol	Source	Source Port Range	Destination	Destination Port Range
Default rule	*	Deny	All	0.0.0.0/0	All	0.0.0.0/0	All

Inbound rules of Fw-B							
Inbound Rule	Rule Number	Action	Protocol	Source	Source Port Range	Destination	Destination Port Range
B01 (Custom)	1	Allow	TCP	10.1.0.0/24	All	10.0.1.0/24	22
Default rule	*	Deny	All	0.0.0.0/0	All	0.0.0.0/0	All
Outbound rules of Fw-B							
Outbound Rule	Rule Number	Action	Protocol	Source	Source Port Range	Destination	Destination Port Range
B02 (Custom)	1	Allow	ICMP	10.0.1.0/24	All	10.1.0.0/24	All
Default rule	*	Deny	All	0.0.0.0/0	All	0.0.0.0/0	All

NOTE

The above figure shows how network ACLs control traffic in and out of subnets. In actual services, the security groups control traffic from and to the instances associated with it. For details about network ACLs and security groups, see **What Is Access Control?**

Network ACL Rules

- Network ACL has inbound and outbound rules that are used to control traffic in and out of subnets.
  - Inbound rules: control traffic sent to the instances in a subnet.
  - Outbound rules: control traffic from the instances in a subnet to external networks.
- You need to define the protocol, source and destination ports, source and destination IP addresses, and other information for network ACL rules.
  - Rule number: Network ACL rules are matched in ascending order, from the lowest to highest rule number.

The default network ACL rule is marked with an asterisk (*) and is the very last rule that will be used for matching.

- **Status**: **Enabled** or **Disabled**. Enabled rules are applied, while disabled rules are not.
- **Type**: **IPv4** or **IPv6**.
- **Action**: **Allow** or **Deny**. If a request matches a network ACL rule, the action defined in the rule is taken to allow or deny the request.
- **Protocol**: The protocol to match traffic. The value can be **TCP**, **UDP**, or **ICMP**.
- **Source**/**Destination**: The source or destination of the traffic.

  The source or destination can be an IP address, CIDR block, or IP address group.

  - **IP address**: an IPv4/IPv6 address, an IPv4/IPv6 CIDR block, for example, 192.168.10.10/32 (IPv4 address), 192.168.1.0/24 (IPv4 CIDR block), or 2407:c080:802:469::/64 (IPv6 CIDR block).
  - **IP address group**: You can add multiple IP addresses with the same security requirements to an **IP address group** and select this IP address group when you configure a rule.
- **Source Port Range**/**Destination Port Range**: The source or destination ports or port ranges. The value ranges from 1 to 65535.

How Network ACL Rules Work

- After a network ACL is created, you can associate it with one or more subnets to control traffic in and out of the subnets. You can associate a network ACL with multiple subnets. However, a subnet can only be associated with one network ACL.
- Network ACLs are stateful. If the network ACL rule allows outbound traffic from your instance, the response to the outbound traffic is allowed to flow in, regardless of the inbound rule settings. Similarly, if inbound traffic is allowed, responses to such inbound traffic are allowed to flow out, regardless of the outbound rule settings.
- Network ACLs use connection tracking to track traffic to and from instances. Changes to inbound and outbound rules do not take effect immediately for the existing traffic.

  If you add, modify, or delete a network ACL rule, or associate or disassociate a subnet with or from a network ACL, all the inbound and outbound persistent connections will not be disconnected. New rules will only be applied for the new connections.

NOTICE

After a persistent connection is disconnected, new connections will not be established immediately until the timeout period of connection tracking expires. For example, after an ICMP persistent connection is disconnected, a new connection will be established and a new rule will be applied when the timeout period (30s) expires.

- The timeout period of connection tracking varies by protocol. The timeout period of a TCP connection in the established state is 600s, and that of an ICMP connection is 30s. For other protocols, if packets are received in both inbound and outbound directions, the connection tracking timeout period is 180s. If packets are received only in one direction, the connection tracking timeout period is 30s.
- The timeout period of TCP connections varies by connection status. The timeout period of a TCP connection in the established state is 600s, and that of a TCP connection in the FIN-WAIT state is 30s.

- Each network ACL has the default inbound and outbound rules, as shown in **Table 6-41**. If a network ACL has no custom rules, the default inbound and outbound rules are applied, denying all traffic in and out of a subnet. You can use the default rules only when there is no need for traffic to go in and out of a subnet. If the traffic needs to go in and out of the subnet, you need to add custom rules to control traffic as required.

  Table 6-41 Default network ACL rules

Direction	Rule Number	Action	Protocol	Source	Source Port Range	Destination	Destination Port Range
Inbound	*	Deny	All	0.0.0.0/0	All	0.0.0.0/0	All
Outbound	*	Deny	All	0.0.0.0/0	All	0.0.0.0/0	All

- The default and custom rules of a network ACL does not block the traffic described in **Table 6-42**.

  Table 6-42 Traffic not blocked by network ACL rules

Direction	Description
Inbound	Traffic between the source and destination in the same subnet
	Broadcast traffic to 255.255.255.255/32
	Multicast traffic to 224.0.0.0/24
Outbound	Traffic between the source and destination in the same subnet

Direction	Description
	Broadcast traffic to 255.255.255.255/32
	Multicast traffic to 224.0.0.0/24
	TCP metadata traffic to 169.254.169.254/32 over port 80
	Traffic to 100.125.0.0/16 that is reserved for public services on the cloud, such as the DNS server address and NTP server address

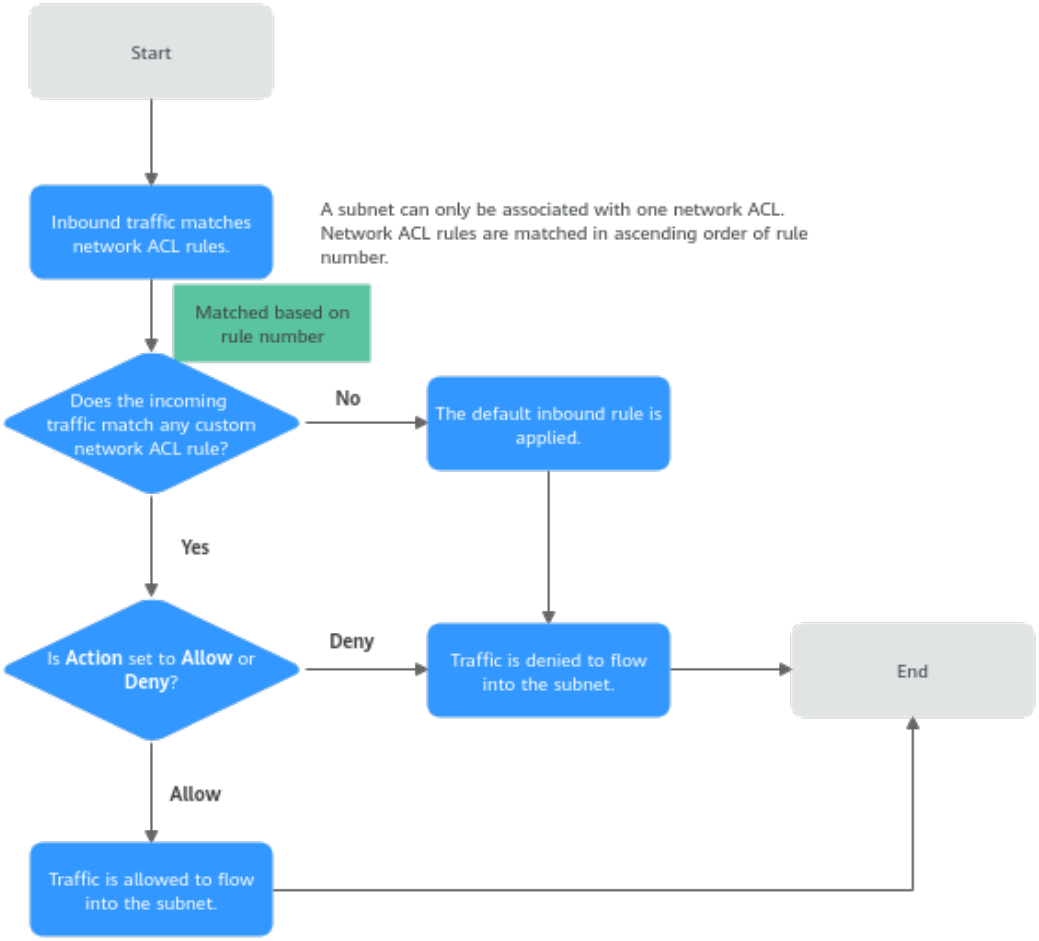
How Traffic Matches Network ACL Rules

A subnet can be associated with only one network ACL. If there are multiple rules on the network ACL, rules are matched in ascending order, from the lowest to highest rule number. The default network ACL rule is marked with an asterisk (*) and is the very last rule that will be used for matching.

The matching sequence of inbound traffic is the same as that of outbound traffic. The following takes inbound traffic as an example to describe how the rules are applied.

- If a custom rule is matched:
  - If **Action** is set to **Deny**, traffic is denied to flow into the subnet.
  - If **Action** is set to **Allow**, traffic is allowed to flow into the subnet.
- If no custom rule is matched, the default rule is applied, denying traffic to flow into the subnet.

Figure 6-27 Network ACL matching



How Network ACLs Are Used

A network ACL controls traffic in and out of a subnet. If both security group and network ACL rules are configured, traffic is matched against network ACL rules first and then security group rules. You can add security group rules as required and use network ACLs as an additional layer of protection for your subnets. The following provides some examples on how network ACLs can be used.

Controlling External Access to Instances in a Subnet

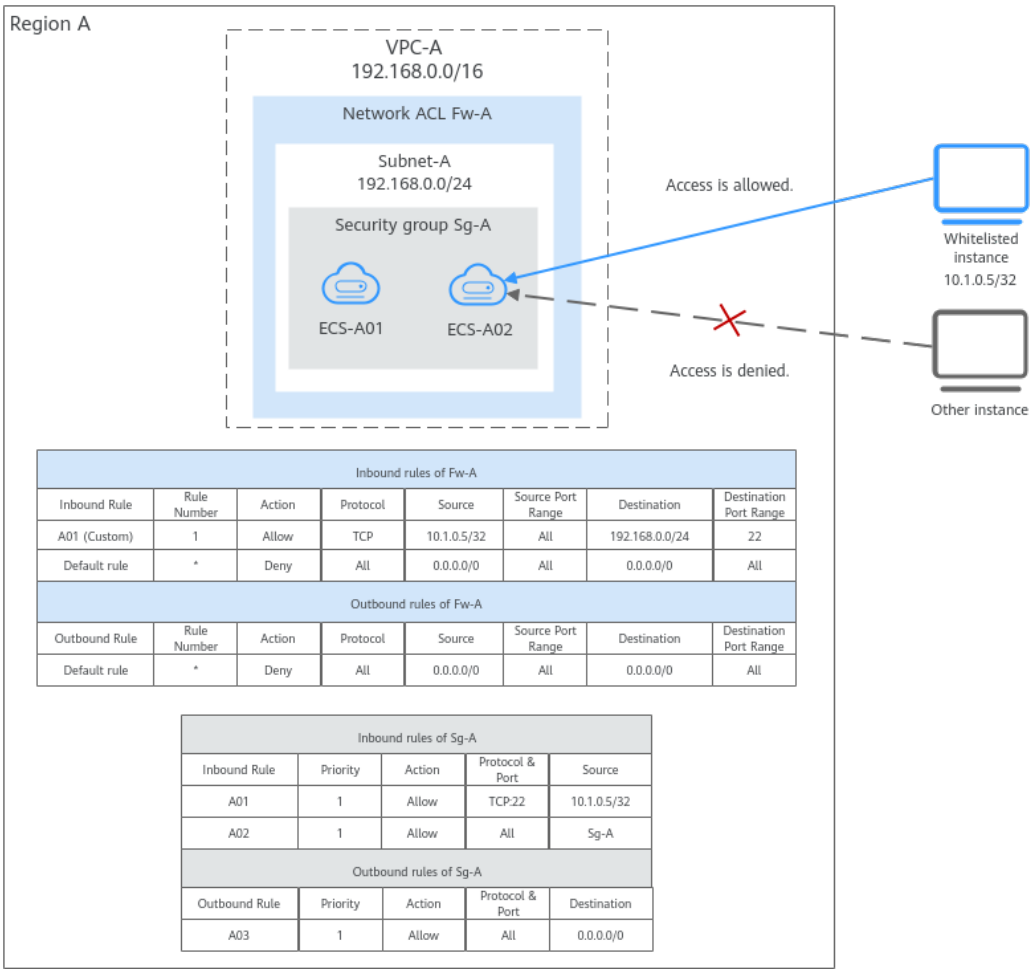
In Figure 6-28, **ECS-A01** and **ECS-A02** in **Subnet-A** need to communicate with each other, and the instance with the IP address **10.1.0.5/32** needs to be whitelisted to allow it to remotely log in to **ECS-A01** and **ECS-A02** to perform O&M operations. The whitelisted instance can be a local PC, an instance in a different subnet of **VPC-A**, or an instance in another VPC. You need to configure network ACL and security group rules to allow the whitelisted instance to access ECSs in **VPC-A** and deny any other traffic.

- Network ACL rules:
  - Inbound rule: Custom rule **A01** allows the whitelisted instance to remotely log in to the instances in **Subnet-A** over SSH. The default rule denies any other traffic to the subnet.

  - Outbound rule: Network ACLs are stateful. The responses to inbound requests are allowed to leave the subnet. This means you do not need to additionally add outbound rules to allow such response traffic. The default rule denies any other outbound traffic.

- Security group rules:
  - Inbound rule: Rule **A01** allows the whitelisted instance to remotely log in to instances in **Subnet-A** over SSH. Rule **A02** allows instances in the security group to communicate with each other. Other traffic is denied to access the instances in security group **Sg-A**.
  - Outbound rule: Rule **A03** allows instances in **Sg-A** to access external resources.

Figure 6-28 Controlling external access to instances in a subnet



Inbound rules of Fw-A							
Inbound Rule	Rule Number	Action	Protocol	Source	Source Port Range	Destination	Destination Port Range
A01 (Custom)	1	Allow	TCP	10.1.0.5/32	All	192.168.0.0/24	22
Default rule	*	Deny	All	0.0.0.0/0	All	0.0.0.0/0	All
Outbound rules of Fw-A							
Outbound Rule	Rule Number	Action	Protocol	Source	Source Port Range	Destination	Destination Port Range
Default rule	*	Deny	All	0.0.0.0/0	All	0.0.0.0/0	All

Inbound rules of Sg-A				
Inbound Rule	Priority	Action	Protocol & Port	Source
A01	1	Allow	TCP:22	10.1.0.5/32
A02	1	Allow	All	Sg-A
Outbound rules of Sg-A				
Outbound Rule	Priority	Action	Protocol & Port	Destination
A03	1	Allow	All	0.0.0.0/0

If you set loose security group rules, network ACL rules can add an additional layer of protection. As described in **Table 6-43**, the security group rule allows any IP address to remotely log in to instances in the security group. The inbound rule of **Fw-A** associated with **Subnet-A** allows only the specified IP address (10.1.0.5/32) to access instances in **Subnet-A**. The default rule denies other traffic to the subnet, eliminating possible security risks.

Table 6-43 Security group rules

Direction	Priority	Action	Type	Protocol & Port	Source	Description
Inbound	1	Allow	IPv4	TCP:22	IP address: 0.0.0.0/0	Allows any IP address to remotely log in to instances in the security group using SSH

NOTE

For more network ACL examples, see **Network ACL Configuration Examples**.

Controlling Communications Between Instances in Different Subnets

In **Figure 6-29**, **VPC-X** has two subnets: **Subnet-X01** and **Subnet-X02**. **ECS-01** and **ECS-02** work in **Subnet-X01**, and **ECS-03** works in **Subnet-X02**. Suppose you want to:

- Connect **ECS-02** to **ECS-03**.
- Isolate **ECS-01** from **ECS-03**.

To achieve this purpose, you need to configure security group and network ACL rules as follows:

1. Add inbound and outbound rules to **Sg-A** to ensure that the ECSs in this security group can communicate with each other.

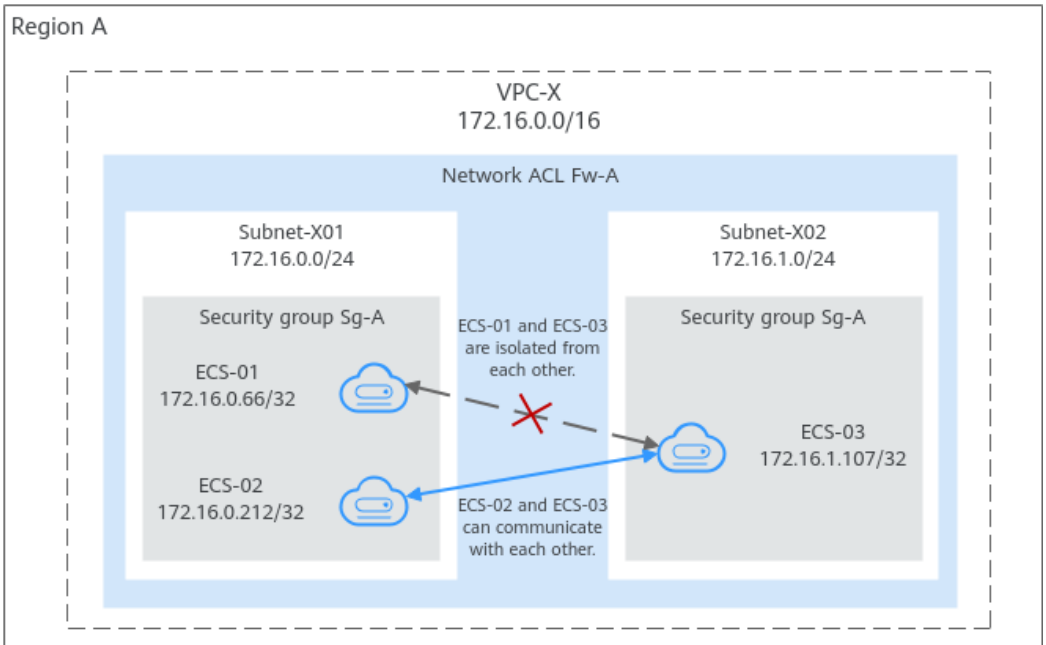
   The subnet has not been associated with a network ACL, so after the security group rules are added, both **ECS-01** and **ECS-02** can communicate with **ECS-03**.

2. Associate **Subnet-X01** and **Subnet-X02** with **Fw-A**.

   If there is only the default rule in **Fw-A**, instances in the same subnet can communicate with each other, while instances in different subnets are isolated from each other. In this case, **ECS-01** and **ECS-02** can communicate with each other, while **ECS-01** and **ECS-03** as well as **ECS-02** and **ECS-03** are isolated from each other.

3. Add custom rules to **Fw-A** to allow **ECS-02** to communicate with **ECS-03**.
   - Add custom rule A01 to allow **ECS-03** to access **Subnet-X01**.
   - Add custom rule A02 to allow **ECS-02** to access **Subnet-X02**.
   - Add custom rule A03 to allow traffic destined for **ECS-03** to leave **Subnet-X01**.
   - Add custom rule A04 to allow traffic destined for **ECS-02** to leave **Subnet-X02**.

Figure 6-29 Controlling communications between instances in different subnets



Inbound rules of Fw-A							
Inbound Rule	Rule Number	Action	Protocol	Source	Source Port Range	Destination	Destination Port Range
A01 (Custom)	1	Allow	All	172.16.1.107/32	All	0.0.0.0/0	All
A02 (Custom)	2	Allow	All	172.16.0.212/32	All	0.0.0.0/0	All
Default rule	*	Deny	All	0.0.0.0/0	All	0.0.0.0/0	All
Outbound rules of Fw-A							
Outbound Rule	Rule Number	Action	Protocol	Source	Source Port Range	Destination	Destination Port Range
A03 (Custom)	1	Allow	All	0.0.0.0/0	All	172.16.1.107/32	All
A04 (Custom)	2	Allow	All	0.0.0.0/0	All	172.16.0.212/32	All
Default rule	*	Deny	All	0.0.0.0/0	All	0.0.0.0/0	All

The inbound rule of Sg-A				
Inbound Rule	Priority	Action	Protocol & Port	Source
A01	1	Allow	All	Sg-A
The outbound rule of Sg-A				
Outbound Rule	Priority	Action	Protocol & Port	Destination
A02	1	Allow	All	Sg-A

NOTE

For more network ACL examples, see **Network ACL Configuration Examples**.

Network ACL Configuration Procedure

Figure 6-30 Procedure for configuring a network ACL

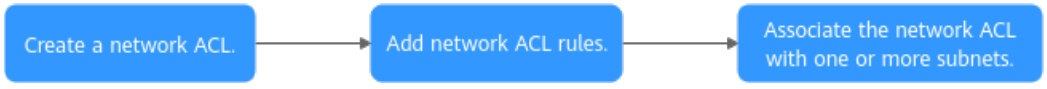


Table 6-44 Procedure for configuring a network ACL

No.	Step	Description	Reference
1	Create a network ACL.	A network ACL comes with default inbound and outbound rules that deny traffic in and out of a subnet. The default rules cannot be deleted or modified.	**Creating a Network ACL**
2	Add inbound and outbound rules.	You can add custom rules to control traffic in and out of a subnet. Traffic will be preferentially matched against the custom rules.	**Adding a Network ACL Rule**
3	Associate the network ACL with one or more subnets.	You can associate the network ACL with one or more subnets. If it is enabled, it controls traffic in and out of the subnets. A subnet can be associated with only one network ACL.	**Associating Subnets with a Network ACL**

Constraints on Using Network ACLs

- By default, each account can have up to 200 network ACLs in a region.
- A network ACL can have no more than 100 rules in one direction, or performance will deteriorate.
- For each network ACL rule, up to 124 rules can have IP address groups associated in either inbound or outbound direction.
- For inbound network ACL rules, the sum of the rules with **Source** set to **IP address group**, of the rules with **Destination** set to **IP address group**, of the rules with **Source Port Range** set to inconsecutive ports, and of the rules **Destination Port Range** set to inconsecutive ports, cannot exceed 120. If there are both IPv4 and IPv6 network ACL rules, up to 120 rules can be added for each type.

  The limits on outbound network ACL rules are the same as those on inbound network ACL rules.

  For example, to add inbound IPv4 rules to a network ACL, you can refer to **Table 6-45** for rules that meet the restrictions. Of these rules, rule A02 uses inconsecutive ports (22-24,25) as the source port range and IP address group ipGroup-A as the source. In this case, only one quota is occupied.

Table 6-45 Inbound network ACL rules

Rule No.	Rule Number	Type	Action	Protocol	Source	Source Port Range	Destination	Destination Port Range
Rule A01	1	IPv4	Deny	TCP	0.0.0.0/0	22,25,27	0.0.0.0/0	1-65535
Rule A02	2	IPv4	Allow	TCP	**IP address group: ipGroup-A**	**22-24,25**	0.0.0.0/0	1-65535
Rule A03	3	IPv4	Allow	All	0.0.0.0/0	All	IP address group: ipGroup-B	All
Rule A04	4	IPv4	Allow	UDP	0.0.0.0/0	1-65535	0.0.0.0/0	80-83,87

- Traffic from load balancers is not restricted by network ACL and security group rules if:

  Transfer Client IP Address is enabled for the listener of a load balancer.

  The load balancer can still forward traffic to backend servers, even if there is a rule that denies traffic from the load balancer to the backend servers.

6.3.2 Network ACL Configuration Examples

You can use network ACLs to control the traffic in and out of a subnet. When both security groups and network ACLs are configured, traffic matches network ACL rules first and then security group rules. You can add security group rules as required and use network ACLs to protect instances in the associated subnets. The following provides some examples on how network ACLs can be used.

- **Denying External Access to a Specific Port in a Subnet**
- **Denying Access from a Specific IP Address**
- **Allowing External Access to Specific Ports on an Instance in a Subnet**

NOTICE

If your network ACL rules do not work, **submit a service ticket**.

Precautions

Note the following before configuring network ACL rules:

- Each network ACL has default rules, as shown in **Table 6-46**. If a network ACL has no custom rules, the default inbound and outbound rules are applied, denying all traffic in and out of a subnet.

  Table 6-46 Default network ACL rules

Direction	Rule Number	Action	Protocol	Source	Source Port Range	Destination	Destination Port Range
Inbound	*	Deny	All	0.0.0.0/0	All	0.0.0.0/0	All
Outbound	*	Deny	All	0.0.0.0/0	All	0.0.0.0/0	All

- You do not need to add a rule to allow response traffic to inbound requests. This is because the network ACLs are stateful and allow the responses to leave the subnet without being controlled by rules.

For more information about how network ACL rules work, see **How Network ACL Rules Work**.

Denying External Access to a Specific Port in a Subnet

If you want to block TCP port 445 to protect instances against WannaCry ransomware attacks, you can add inbound rules described in **Table 6-47** to protect the instances in 10.0.0.0/24.

1. The default rule denies any traffic to the subnet. You need to add custom rule 02 to allow inbound traffic.
2. Add custom rule 01 to deny all inbound traffic to TCP port 445. Place the deny rule above the allow rule to let the deny rule be applied first. For details, see **Adding a Network ACL Rule (Custom Rule Numbers)**.

Table 6-47 Inbound rules for denying external access to a specific port in a subnet

Direction	Rule Number	Type	Action	Protocol	Source	Source Port Range	Destination	Destination Port Range	Description
Inbound	1	IPv4	Deny	TCP	0.0.0.0/0	All	10.0.0.0/24	445	Custom rule 01
Inbound	2	IPv4	Allow	All	0.0.0.0/0	All	10.0.0.0/24	All	Custom rule 02

Direction	Rule Number	Type	Action	Protocol	Source	Source Port Range	Destination	Destination Port Range	Description
Inbound	*	--	Deny	All	0.0.0.0/0	All	0.0.0.0/0	All	Default rule

Denying Access from a Specific IP Address

You can add inbound rules as described in **Table 6-48** to deny the access from abnormal IP addresses, for example, 10.1.1.12/32, to protect the instances in 10.5.0.0/24.

1. The default rule denies any traffic to the subnet. You need to add custom rule 02 to allow inbound traffic.
2. Add custom rule 01 to deny traffic from 10.1.1.12/32 to 10.5.0.0/24. Place the deny rule above the allow rule to let the deny rule be applied first. For details, see **Adding a Network ACL Rule (Custom Rule Numbers)**.

Table 6-48 Inbound rules for denying access from a specific IP address

Direction	Rule Number	Type	Action	Protocol	Source	Source Port Range	Destination	Destination Port Range	Description
Inbound	1	IPv4	Deny	TCP	10.1.1.12/32	All	10.5.0.0/24	All	Custom rule 01
Inbound	2	IPv4	Allow	All	0.0.0.0/0	All	10.5.0.0/24	All	Custom rule 02
Inbound	*	--	Deny	All	0.0.0.0/0	All	0.0.0.0/0	All	Default rule

Allowing External Access to Specific Ports on an Instance in a Subnet

If you deploy a web server in a subnet and want this server to be accessible from the Internet, you need to add network ACL and security group rule to allow HTTP traffic over port 80 and HTTPS traffic over port 443.

1. Add network ACL rules listed in **Table 6-49**.
   - Add custom rule A01 to allow any HTTP traffic to the instance in the subnet (10.8.0.0/24) over port 80.

– Add custom rule A02 to allow any HTTPS traffic to the instance in the subnet (10.8.0.0/24) over port 443.

Table 6-49 Network ACL rules for allowing access to specific ports on an instance in a subnet

Direction	Rule Number	Type	Action	Protocol	Source	Source Port Range	Destination	Destination Port Range	Description
Inbound	1	IPv4	Allow	TCP	0.0.0.0/0	All	10.8.0.0/24	80	Custom rule 01
Inbound	2	IPv4	Allow	TCP	0.0.0.0/0	All	10.8.0.0/24	443	Custom rule 02
Inbound	*	--	Deny	All	0.0.0.0/0	All	0.0.0.0/0	All	Default rule
Outbound	*	--	Deny	All	0.0.0.0/0	All	0.0.0.0/0	All	Default rule

2. Add security group rules listed in **Table 6-50**.
   – Add inbound rule 01 to allow any HTTP traffic to the instance over port 80.
   – Add inbound rule 02 to allow any HTTPS traffic to the instance over port 443.
   – Add outbound rule 03 to allow any traffic to leave the security group.

     You do not need to worry about the loose control of the security group outbound rules. Network ACL rules only allow response traffic to inbound requests to leave the subnet.

Table 6-50 Security group rules for allowing access to specific ports

Direction	Priority	Action	Type	Protocol & Port	Source/ Destination	Description
Inbound	1	Allow	IPv4	TCP: 80	IP address: 0.0.0.0/0	Rule 01
Inbound	1	Allow	IPv4	TCP: 443	IP address: 0.0.0.0/0	Rule 02

Direction	Priority	Action	Type	Protocol & Port	Source/ Destination	Description
Outbound	1	Allow	IPv4	All	IP address: 0.0.0.0/0	Rule 03

6.3.3 Managing Network ACLs

6.3.3.1 Creating a Network ACL

Scenarios

A security group protects the instances in it, such as ECSs, databases, and containers, while a network ACL protects associated subnets and all the instances in the subnets. Security groups are mandatory, while network ACLs are optional. If you want to add an additional layer of protection, you can create a network ACL and associate it with one or more subnets. Network ACLs and security groups can be used together for fine-grained and comprehensive access control.

Procedure

1. Go to the **network ACL list page**.
2. In the upper right corner of the network ACL list, click **Create Network ACL**.
3. On the displayed page, configure the parameters as prompted.

Table 6-51 Parameter descriptions

Parameter	Description	Example Value
Region	Mandatory A network ACL can only be associated with the subnets in the same region.	-
Name	Mandatory The network ACL name. The name can contain a maximum of 64 characters, which may consist of letters, digits, underscores (_), hyphens (-), and periods (.). The name cannot contain spaces.	fw-A

Parameter	Description	Example Value
Enterprise Project	Mandatory Enterprise project that the network ACL belongs to. An enterprise project facilitates project-level management and grouping of cloud resources and users. The default project is **default**. For details about creating and managing enterprise projects, see the **Enterprise Management User Guide**.	default
Tag (Optional)	Optional When creating a network ACL, you can add tags to it to help you identify and search for given network ACLs. For details, see **Managing Network ACL Tags**.	**Tag key**: test **Tag value**: 01
Description (Optional)	Supplementary information about the network ACL. This parameter is optional. The description can contain a maximum of 255 characters and cannot contain angle brackets (< or >).	N/A

4. Click **Create Now**.

Follow-up Operations

1. A network ACL comes with default inbound and outbound rules that deny all traffic in and out of associated subnets. You can add custom rules to allow traffic by referring to **Adding a Network ACL Rule**. Traffic will preferentially match the custom rules.
2. You need to associate the enabled network ACL with the subnets by referring to **Associating Subnets with a Network ACL**.

6.3.3.2 Modifying a Network ACL

Scenarios

You can modify the name and description of a network ACL.

Procedure

1. Log in to the management console.
2. Click  in the upper left corner and select the desired region and project.

3. Click ☰ in the upper left corner and choose **Networking** > **Virtual Private Cloud**.

   The **Virtual Private Cloud** page is displayed.
4. In the navigation pane on the left, choose **Access Control** > **Network ACLs**.

   The network ACL list is displayed.
5. In the network ACL list, locate the target network ACL and click its name.

   The network ACL summary page is displayed.
6. On the **Summary** tab, modify the name and description as needed.

6.3.3.3 Enabling or Disabling a Network ACL

Scenarios

- If a network ACL is disabled, custom rules will become invalid but default rules are still applied. As a result, all traffic to and from the associated subnets are denied. If a network ACL has a subnet associated, disabling it will interrupt the network traffic to and from the subnet.
- If a network ACL is enabled, both custom and default rules are applied. If a network ACL has a subnet associated and has only default rules, enabling it will interrupt the network traffic to and from the subnet.

Procedure

1. Log in to the management console.
2. Click  in the upper left corner and select the desired region and project.
3. Click ☰ in the upper left corner and choose **Networking** > **Virtual Private Cloud**.

   The **Virtual Private Cloud** page is displayed.
4. In the navigation pane on the left, choose **Access Control** > **Network ACLs**.

   The network ACL list is displayed.
5. In the network ACL list, enable or disable the target network ACL.
   - Enabling a network ACL
     i. Locate the target network ACL and choose **More** > **Enable** in the **Operation** column.

        A confirmation dialog box is displayed.
     ii. Confirm the information and click **OK**.
   - Disabling a network ACL
     i. Locate the target network ACL and choose **More** > **Disable** in the **Operation** column.

        A confirmation dialog box is displayed.
     ii. Confirm the information and click **OK**.

6.3.3.4 Viewing a Network ACL

Scenarios

You can check the details of a network ACL, such as the name, rules, and associated subnets.

You can search for a network ACL by name, ID, and description.

Procedure

1. Log in to the management console.
2. Click  in the upper left corner and select the desired region and project.
3. Click  in the upper left corner and choose **Networking** > **Virtual Private Cloud**.

   The **Virtual Private Cloud** page is displayed.
4. In the navigation pane on the left, choose **Access Control** > **Network ACLs**.

   The network ACL list is displayed.
5. In the network ACL list, locate the target network ACL and click its name.

   The network ACL summary page is displayed.
6. On the **Summary** tab, you can view the following information:
   - Basic information: name, ID, status, and description.
   - Inbound and outbound rules: rule number, status, protocol, source, source port, destination, and destination port.
   - Associated subnets: the subnets associated with the network ACL. A network ACL can be associated with multiple subnets.
   - Tag: tags of a network ACL.

6.3.3.5 Managing Network ACL Tags

Scenarios

Tags help you identify, classify, and search for network ACLs. You can perform the following operations to manage tags of a network ACL:

- Add a network ACL tag.
- Modify a network ACL tag.
- Delete a network ACL tag.

For details about tag key and value requirements, see **Table 6-52**.

Table 6-52 Network ACL tag key and value requirements

Parameter	Requirements	Example Value
Tag key	• For each resource, each tag key must be unique, and each tag key can only have one tag value. • Cannot be left blank. • Can contain a maximum of 128 characters. • Can contain letters in any language, digits, spaces, underscores (_), periods (.), colons (:), equal signs (=), plus signs (+), minus signs (-), and at signs (@). • Cannot start with _sys_ or a space or end with a space.	test
Tag value	• Can be left blank. • Can contain a maximum of 255 characters. • Can contain letters in any language, digits, spaces, underscores (_), periods (.), colons (:), slashes (/), equal signs (=), plus signs (+), minus signs (-), and at signs (@). • Cannot start or end with a space.	01

Notes and Constraints

Each cloud resource can have a maximum of 20 tags.

Procedure

1. Log in to the management console.
2. Click in the upper left corner and select the desired region and project.
3. Click in the upper left corner and choose **Networking** > **Virtual Private Cloud**.

   The **Virtual Private Cloud** page is displayed.
4. Go to the **network ACL list page**.
5. In the network ACL list, click the name of the target network ACL.

   The network ACL details page is displayed.

6. On the **Tags** tab, click **Edit Tag** in the upper left corner above the tag list.
   The **Edit Tag** dialog box is displayed.
7. Perform the following operations on the tag as required:
   - Adding a tag: Click +, enter a tag key and value, and click **OK**.
   - Modifying a tag: Click × next to the target tag key or value, delete the original value, enter a new value, and click **OK**.
   - Deleting a tag: Click **Delete** next to the target tag and click **OK**.

6.3.3.6 Deleting a Network ACL

Scenarios

You can delete a network ACL when it is no longer required.

Deleting a network ACL will also disassociate it from its associated subnets. Be careful with this operation as it may interrupt services.

Procedure

1. Log in to the management console.
2. Click in the upper left corner and select the desired region and project.
3. Click in the upper left corner and choose **Networking** > **Virtual Private Cloud**.
   The **Virtual Private Cloud** page is displayed.
4. In the navigation pane on the left, choose **Access Control** > **Network ACLs**.
   The network ACL list is displayed.
5. In the network ACL list, locate the target network ACL and choose **More** > **Delete** in the **Operation** column.
   A confirmation dialog box is displayed.
6. Confirm the information and click **OK**.

6.3.4 Managing Network ACL Rules

6.3.4.1 Adding a Network ACL Rule

Scenarios

You can add inbound and outbound rules to a network ACL to control the traffic in and out of a subnet. Network ACL rules are matched in ascending order, either by the system-generated rule numbers or those you define.

- **Adding a Network ACL Rule (Default Rule Numbers)**: Rules are matched in order of their number, starting with the lowest. The rule number is automatically assigned based on the time when the rule is added.
  In **Table 6-53**, there are two custom inbound rules (rule A and rule B) and one default rule. The rule A number is 1 and rule B number is 2. The default rule is the last rule that is used for matching traffic. When you add rule C, the

rule number will be 3, which will be matched later than rules A and B but earlier than the default rule.

Table 6-53 Default rule numbers

Rule Number (Rules A and B)		Rule Number (Rules A, B, and C)	
Custom rule A	1	Custom rule A	1
--	--	Custom rule B	2
Custom rule B	2	**Custom rule C**	**3**
Default rule	*	Default rule	*

- **Adding a Network ACL Rule (Custom Rule Numbers)**: If you want a rule to be matched earlier or later than a specific rule, you can insert the rule above or below the specific rule.

  In **Table 6-54**, there are two custom inbound rules (rule A and rule B) and one default rule. The rule A number is 1 and rule B number is 2. The default rule is the last rule that is used for matching traffic. If you want rule C to be matched earlier than rule B, you can insert rule C above rule B. After rule C is added, the rule C number is 2, and rule B number is 3.

  Table 6-54 Custom rule numbers

Rule Number (Rules A and B)		Rule Number (Rules A, B, and C)	
Custom rule A	1	Custom rule A	1
--	--	**Custom rule C**	**2**
Custom rule B	2	Custom rule B	3
Default rule	*	Default rule	*

Constraints

A network ACL can contain up to 100 rules in one direction, or performance will deteriorate.

Adding a Network ACL Rule (Default Rule Numbers)

1. Log in to the management console.
2. Click in the upper left corner and select the desired region and project.
3. Click in the upper left corner and choose **Networking** > **Virtual Private Cloud**.

   The **Virtual Private Cloud** page is displayed.
4. In the navigation pane on the left, choose **Access Control** > **Network ACLs**.

   The network ACL list is displayed.

5. In the network ACL list, locate the target network ACL and click its name.
   The network ACL summary page is displayed.
6. On the **Inbound Rules** or **Outbound Rules** tab, click **Add Rule**.
   The **Add Inbound Rule** or **Add Outbound Rule** dialog box is displayed.
7. Configure required parameters.
   - Click ⊕ to add more rules.
   - Locate the row that contains the network ACL rule and click **Replicate** in the **Operation** column to replicate an existing rule.

Table 6-55 Parameter descriptions

Parameter	Description	Example Value
Type	Network ACL type. There are two options: • **IPv4** • **IPv6**	IPv4
Action	The action for the network ACL rule. There are two options: • **Allow**: allows matched traffic in and out of a subnet. • **Deny**: denies matched traffic in and out of a subnet.	Allow
Protocol	The protocol supported by the network ACL to match traffic. The value can be **TCP**, **UDP**, or **ICMP**.	TCP

Parameter	Description	Example Value
Source	The source from which the traffic is allowed or denied. The source can be: • **IP address** – Single IP address: IP address/mask Example IPv4 address: 192.168.10.10/32 Example IPv6 address: 2002:50::44/128 – IP address range in CIDR notation: IP address/mask Example IPv4 address range: 192.168.52.0/24 Example IPv6 address range: 2407:c080:802:469::/64 – All IP addresses 0.0.0.0/0 represents all IPv4 addresses. ::/0 represents all IPv6 addresses. • **IP address group**: The source is an IP address group. An IP address group is a collection of one or more IP addresses. You can select an available IP address group from the drop-down list. An IP address group can help you manage IP address ranges and IP addresses with same security requirements in an easier way. Either the source or the destination of a network ACL rule can use the IP address group. For example, if the source uses an IP address group, the destination address cannot use an IP address group. If no IP address groups are available, create one by referring to **Creating an IP Address Group**.	192.168.0.0/24

Parameter	Description	Example Value
Source Port Range	The source ports or port ranges used to match traffic. The value ranges from 1 to 65535. Enter ports in the following format: ● Individual port: Enter a port, such as **22**. ● Consecutive ports: Enter a port range, such as **22-30**. ● Non-consecutive ports: Enter ports and port ranges, such as **22,24-30**. You can enter a maximum of 20 ports and port ranges. Each port range must be unique. ● All ports: Leave it empty or enter **1-65535**.	22-30

Parameter	Description	Example Value
Destination	The destination to which the traffic is allowed or denied. The destination can be: • **IP address** – Single IP address: IP address/mask Example IPv4 address: 192.168.10.10/32 Example IPv6 address: 2002:50::44/128 – IP address range in CIDR notation: IP address/mask Example IPv4 address range: 192.168.52.0/24 Example IPv6 address range: 2407:c080:802:469::/64 – All IP addresses 0.0.0.0/0 represents all IPv4 addresses. ::/0 represents all IPv6 addresses. • **IP address group**: The destination is an IP address group. An IP address group is a collection of one or more IP addresses. You can select an available IP address group from the drop-down list. An IP address group can help you manage IP address ranges and IP addresses with same security requirements in an easier way. Either the source or the destination of a network ACL rule can use the IP address group. For example, if the source uses an IP address group, the destination address cannot use an IP address group. If no IP address groups are available, create one by referring to **Creating an IP Address Group**.	0.0.0.0/0

Parameter	Description	Example Value
Destination Port Range	The destination ports or port ranges used to match traffic. The value ranges from 1 to 65535. Enter ports in the following format: ● Individual port: Enter a port, such as **22**. ● Consecutive ports: Enter a port range, such as **22-30**. ● Non-consecutive ports: Enter ports and port ranges, such as **22,23-30**. You can enter a maximum of 20 ports and port ranges. Each port range must be unique. ● All ports: Leave it empty or enter **1-65535**.	22-30
Description	Supplementary information about the network ACL rule. This parameter is optional. The description can contain a maximum of 255 characters and cannot contain angle brackets (< or >).	N/A

8. Click **OK**.

   Return to the rule list to check the new rule.

   – Rules are assigned a number based on the order they are added, with lower-numbered rule matched earlier.
   – If the status of the new rule is **Enabled**, the rule is applied.

Adding a Network ACL Rule (Custom Rule Numbers)

1. Log in to the management console.
2. Click  in the upper left corner and select the desired region and project.
3. Click  in the upper left corner and choose **Networking** > **Virtual Private Cloud**.

   The **Virtual Private Cloud** page is displayed.
4. In the navigation pane on the left, choose **Access Control** > **Network ACLs**.

   The network ACL list is displayed.
5. In the network ACL list, locate the target network ACL and click its name.

   The network ACL summary page is displayed.
6. Click the **Inbound Rules** or **Outbound Rules** tab and insert a rule.

   – Locate the target rule and choose **More** > **Insert Rule Above** in the **Operation** column. The new rule will be matched earlier than the current rule.

- Locate the target rule and choose **More** > **Insert Rule Below** in the **Operation** column. The new rule will be matched later than the current rule.

6.3.4.2 Modifying a Network ACL Rule

Scenarios

If a network ACL rule no longer meets your requirements, you can modify the port, protocol, and source/destination it.

Modifying rules may affect how and where traffic is directed. Be careful with this operation as it may interrupt services.

Notes and Constraints

Default network ACL rules cannot be modified or deleted.

Procedure

1. Log in to the management console.
2. Click in the upper left corner and select the desired region and project.
3. Click in the upper left corner and choose **Networking** > **Virtual Private Cloud**.

   The **Virtual Private Cloud** page is displayed.
4. In the navigation pane on the left, choose **Access Control** > **Network ACLs**.

   The network ACL list is displayed.
5. In the network ACL list, locate the target network ACL and click its name.

   The network ACL summary page is displayed.
6. Click the **Inbound Rules** or **Outbound Rules** tab, locate the target rule, click **Modify** in the **Operation** column, and modify parameters based on **Table 6-56**.

Table 6-56 Parameter descriptions

Parameter	Description	Example Value
Type	Network ACL type. There are two options: • **IPv4** • **IPv6**	IPv4
Action	The action for the network ACL rule. There are two options: • **Allow**: allows matched traffic in and out of a subnet. • **Deny**: denies matched traffic in and out of a subnet.	Allow

Parameter	Description	Example Value
Protocol	The protocol supported by the network ACL to match traffic. The value can be **TCP**, **UDP**, or **ICMP**.	TCP
Source	The source from which the traffic is allowed or denied. The source can be: • **IP address** – Single IP address: IP address/mask Example IPv4 address: 192.168.10.10/32 Example IPv6 address: 2002:50::44/128 – IP address range in CIDR notation: IP address/mask Example IPv4 address range: 192.168.52.0/24 Example IPv6 address range: 2407:c080:802:469::/64 – All IP addresses 0.0.0.0/0 represents all IPv4 addresses. ::/0 represents all IPv6 addresses. • **IP address group**: The source is an IP address group. An IP address group is a collection of one or more IP addresses. You can select an available IP address group from the drop-down list. An IP address group can help you manage IP address ranges and IP addresses with same security requirements in an easier way. Either the source or the destination of a network ACL rule can use the IP address group. For example, if the source uses an IP address group, the destination address cannot use an IP address group. If no IP address groups are available, create one by referring to **Creating an IP Address Group**.	192.168.0.0/24

Parameter	Description	Example Value
Source Port Range	The source ports or port ranges used to match traffic. The value ranges from 1 to 65535. Enter ports in the following format: ● Individual port: Enter a port, such as **22**. ● Consecutive ports: Enter a port range, such as **22-30**. ● Non-consecutive ports: Enter ports and port ranges, such as **22,24-30**. You can enter a maximum of 20 ports and port ranges. Each port range must be unique. ● All ports: Leave it empty or enter **1-65535**.	22-30

Parameter	Description	Example Value
Destination	The destination to which the traffic is allowed or denied. The destination can be: • **IP address** – Single IP address: IP address/mask Example IPv4 address: 192.168.10.10/32 Example IPv6 address: 2002:50::44/128 – IP address range in CIDR notation: IP address/mask Example IPv4 address range: 192.168.52.0/24 Example IPv6 address range: 2407:c080:802:469::/64 – All IP addresses 0.0.0.0/0 represents all IPv4 addresses. ::/0 represents all IPv6 addresses. • **IP address group**: The destination is an IP address group. An IP address group is a collection of one or more IP addresses. You can select an available IP address group from the drop-down list. An IP address group can help you manage IP address ranges and IP addresses with same security requirements in an easier way. Either the source or the destination of a network ACL rule can use the IP address group. For example, if the source uses an IP address group, the destination address cannot use an IP address group. If no IP address groups are available, create one by referring to **Creating an IP Address Group**.	0.0.0.0/0

Parameter	Description	Example Value
Destination Port Range	The destination ports or port ranges used to match traffic. The value ranges from 1 to 65535. Enter ports in the following format: • Individual port: Enter a port, such as **22**. • Consecutive ports: Enter a port range, such as **22-30**. • Non-consecutive ports: Enter ports and port ranges, such as **22,23-30**. You can enter a maximum of 20 ports and port ranges. Each port range must be unique. • All ports: Leave it empty or enter **1-65535**.	22-30
Description	Supplementary information about the network ACL rule. This parameter is optional. The description can contain a maximum of 255 characters and cannot contain angle brackets (< or >).	N/A

7. Click **OK**.

6.3.4.3 Enabling or Disabling One or More Network ACL Rules

Scenarios

After a rule is added, it is in the **Enabled** status. You can disable it if you need.

- If all custom rules are disabled, they will become invalid but default rules are still applied. As a result, all traffic to and from the associated subnets is denied. Disabling all custom rules may interrupt network traffic. Be careful with this operation as it may interrupt services.
- If a custom rule is enabled, it is applied. Enabling custom rules may affect how and where traffic is directed. Be careful with this operation as it may interrupt services.

To enable or disable a network ACL rule, see **Enabling or Disabling a Network ACL Rule**.

To enable or disable multiple network ACL rules at a time:

- If there is small number of rules to be enabled or disabled, you can select these rules in the network ACL rule list on the console by referring to **Enabling or Disabling Multiple Network ACL Rules Directly on the Console**.
- If there is a large number of rules to be enabled or disabled, you can **export the rule list to a local Excel file**, only keep the rules you want to enable or

disable, and import the file to the console. The system then selects the rules to be processed based on the imported file. For details, see **Enabling or Disabling Multiple Network ACL Rules Using an Excel File**.

Notes and Constraints

Default network ACL rules cannot be modified or deleted.

Enabling or Disabling a Network ACL Rule

1. Log in to the management console.
2. Click  in the upper left corner and select the desired region and project.
3. Click  in the upper left corner and choose **Networking** > **Virtual Private Cloud**.

   The **Virtual Private Cloud** page is displayed.
4. In the navigation pane on the left, choose **Access Control** > **Network ACLs**.

   The network ACL list is displayed.
5. In the network ACL list, locate the target network ACL and click its name.

   The network ACL summary page is displayed.
6. Click the **Inbound Rules** or **Outbound Rules** tab as required.

   The network ACL rule list is displayed.
7. In the rule list, perform the following operations to enable or disable a rule:
   - Enabling a network ACL rule
     i. Locate the target network ACL rule and choose **More** > **Enable** in the **Operation** column.

        A confirmation dialog box is displayed.
     ii. Confirm the information and click **OK**.
   - Disabling a network ACL rule
     i. Locate the target network ACL rule and choose **More** > **Disable** in the **Operation** column.

        A confirmation dialog box is displayed.
     ii. Confirm the information and click **OK**.

Enabling or Disabling Multiple Network ACL Rules Directly on the Console

1. Log in to the management console.
2. Click  in the upper left corner and select the desired region and project.
3. Click  in the upper left corner and choose **Networking** > **Virtual Private Cloud**.

   The **Virtual Private Cloud** page is displayed.
4. In the navigation pane on the left, choose **Access Control** > **Network ACLs**.

   The network ACL list is displayed.
5. In the network ACL list, locate the target network ACL and click its name.

The network ACL summary page is displayed.

6. Click the **Inbound Rules** or **Outbound Rules** tab as required.

   The network ACL rule list is displayed.

7. In the network ACL rule list, select multiple rules.
8. In the rule list, perform the following operations:
   - Enabling multiple network ACL rules at a time
     i. In the upper part corner above the network ACL rule list, choose **More** > **Enable**.

        A confirmation dialog box is displayed.

     ii. Confirm the information and click **OK**.
   - Disabling multiple network ACL rules at a time
     i. In the upper part corner above the network ACL rule list, choose **More** > **Disable**.

        A confirmation dialog box is displayed.

     ii. Confirm the information and click **OK**.

Enabling or Disabling Multiple Network ACL Rules Using an Excel File

1. Log in to the management console.
2. Click  in the upper left corner and select the desired region and project.
3. Click  in the upper left corner and choose **Networking** > **Virtual Private Cloud**.

   The **Virtual Private Cloud** page is displayed.

4. In the navigation pane on the left, choose **Access Control** > **Network ACLs**.

   The network ACL list is displayed.

5. In the network ACL list, locate the target network ACL and click its name.

   The network ACL summary page is displayed.

6. Click the **Inbound Rules** or **Outbound Rules** tab as required.

   The network ACL rule list is displayed.

7. In the upper left corner above the network ACL rule list, click **Batch Operations**.

   The **Batch Operations** dialog box is displayed.

8. Select either of the following methods:
   - Method 1: Click **Download Template** to download the Excel file to your local PC and fill in the network ACL rules to be enabled or disabled in the file.
   - Method 2: **Export the existing rules to a local Excel file**, filter the target rules and keep them as they are, and save the file.

   After the Excel file is ready, take step **9**. The system then automatically selects the target rules based on the imported file.

9. In the **Batch Operations** dialog box, click **Select File**.

The system starts to match the rules in the Excel file against existing network ACL rules based on the type, action, protocol, source, source port range, destination, and destination port range.

- If a rule in the Excel file matches an existing network ACL rule, **Verified** is displayed in the **Result** column. Only the matched rules can be enabled or disabled.
- If a rule fails to be matched, the causes will be displayed in the **Result** column. The possible causes are as follows:
  - There is no such rule in this network ACL.
  - Inconsistent rule direction. For example, you perform the operation on outbound rules on the **Inbound Rules** tab, or the other way around.
  - Duplicate rules in the Excel file. The system automatically filters out the duplicate rules.

10. Confirm the rules and click **OK**.

    The network ACL rule list page is displayed and the target rules are selected automatically.

11. In the rule list, perform the following operations:
    - Enabling multiple network ACL rules at a time
      i. In the upper part corner above the network ACL rule list, choose **More** > **Enable**.

         A confirmation dialog box is displayed.

      ii. Confirm the information and click **OK**.
    - Disabling multiple network ACL rules at a time
      i. In the upper part corner above the network ACL rule list, choose **More** > **Disable**.

         A confirmation dialog box is displayed.

      ii. Confirm the information and click **OK**.

6.3.4.4 Exporting and Importing Network ACL Rules

Scenarios

You can specify rule parameters in an Excel file and import it into an existing network ACL. You can also export rules of a network ACL to an Excel file.

You can import or export network ACL rules if you want to:

- Back up these rules to a local directory as an Excel file.
- Quickly add and restore rules by modifying and importing the Excel file you have exported.
- Quickly add rules to other network ACLs.
- Modify rules in batches. You can export rules as an Excel file, modify these rules in the Excel file, and import the file to the network ACL.

Notes and Constraints

- For optimal performance, you can import or export up to 40 network ACL inbound and outbound at a time.
- Importing rules will not delete existing rules.
- Importing duplicate rules will fail.
- Default rules cannot be exported.

Procedure

1. Log in to the management console.
2. Click  in the upper left corner and select the desired region and project.
3. Click  in the upper left corner and choose **Networking** > **Virtual Private Cloud**.

   The **Virtual Private Cloud** page is displayed.
4. In the navigation pane on the left, choose **Access Control** > **Network ACLs**.

   The network ACL list is displayed.
5. In the network ACL list, locate the target network ACL and click its name.

   The network ACL summary page is displayed.
6. Export or import network ACL rules.
   - Click **Export Rule** to export the network ACL rules to an Excel file.
   - Click **Import Rule** to import the network ACL rules from an Excel file into the current network ACL.

6.3.4.5 Deleting One or More Network ACL Rules

Scenarios

You can delete network ACL rules if you no longer need them.

Deleting rules may affect how and where traffic is directed. Be careful with this operation as it may interrupt services.

To delete a network ACL rule, see **Deleting a Network ACL Rule**.

To delete multiple network ACL rules at a time:

- If there are a small number of rules to be deleted, you can select these rules in the network ACL rule list on the console by referring to **Deleting Network ACL Rules in Batches (Manually Selecting Rules on the Console)**.
- If there are a large number of rules to be deleted, you can **export the rule list to a local Excel file**, only keep the rules you want to delete, and import the file to the console. The system then selects the rules to be processed based on the imported file. For details, see **Deleting Network ACL Rules in Batches (Using an Excel File)**.

Notes and Constraints

Default network ACL rules cannot be modified or deleted.

A maximum of 50 network ACL rules can be deleted at a time.

Deleting a Network ACL Rule

1. Log in to the management console.
2. Click in the upper left corner and select the desired region and project.
3. Click in the upper left corner and choose **Networking** > **Virtual Private Cloud**.

   The **Virtual Private Cloud** page is displayed.
4. In the navigation pane on the left, choose **Access Control** > **Network ACLs**.

   The network ACL list is displayed.
5. In the network ACL list, locate the target network ACL and click its name.

   The network ACL summary page is displayed.
6. Click the **Inbound Rules** or **Outbound Rules** tab as required.

   The network ACL rule list is displayed.
7. In the network ACL rule list, locate the target rule and click **Delete** in the **Operation** column.

   A confirmation dialog box is displayed.
8. Confirm the information and click **OK**.

Deleting Network ACL Rules in Batches (Manually Selecting Rules on the Console)

1. Log in to the management console.
2. Click in the upper left corner and select the desired region and project.
3. Click in the upper left corner and choose **Networking** > **Virtual Private Cloud**.

   The **Virtual Private Cloud** page is displayed.
4. In the navigation pane on the left, choose **Access Control** > **Network ACLs**.

   The network ACL list is displayed.
5. In the network ACL list, locate the target network ACL and click its name.

   The network ACL summary page is displayed.
6. Click the **Inbound Rules** or **Outbound Rules** tab as required.

   The network ACL rule list is displayed.
7. In the network ACL rule list, select multiple rules and click **Delete** in the upper left corner above the list.

   A confirmation dialog box is displayed.
8. Confirm the information and click **OK**.

Deleting Network ACL Rules in Batches (Using an Excel File)

1. Log in to the management console.

2. Click  in the upper left corner and select the desired region and project.
3. Click  in the upper left corner and choose **Networking** > **Virtual Private Cloud**.

   The **Virtual Private Cloud** page is displayed.
4. In the navigation pane on the left, choose **Access Control** > **Network ACLs**.

   The network ACL list is displayed.
5. In the network ACL list, locate the target network ACL and click its name.

   The network ACL summary page is displayed.
6. Click the **Inbound Rules** or **Outbound Rules** tab as required.

   The network ACL rule list is displayed.
7. In the upper left corner above the network ACL rule list, click **Batch Operations**.

   The **Batch Operations** dialog box is displayed.
8. Select either of the following methods:
   - Method 1: Click **Download Template** to download the Excel file to your local PC and fill in the network ACL rules to be enabled or disabled in the file.
   - Method 2: **Export the existing rules to a local Excel file**, filter the target rules and keep them as they are, and save the file.

   After the Excel file is ready, take step **9**. The system then automatically selects the target rules based on the imported file.
9. In the **Batch Operations** dialog box, click **Select File**.

   The system starts to match the rules in the Excel file against existing network ACL rules based on the type, action, protocol, source, source port range, destination, and destination port range.
   - If a rule in the Excel file matches an existing network ACL rule, **Verified** is displayed in the **Result** column. Only the matched rules can be enabled or disabled.
   - If a rule fails to be matched, the causes will be displayed in the **Result** column. The possible causes are as follows:
     - There is no such rule in this network ACL.
     - Inconsistent rule direction. For example, you perform the operation on outbound rules on the **Inbound Rules** tab, or the other way around.
     - Duplicate rules in the Excel file. The system automatically filters out the duplicate rules.
10. Confirm the rules and click **OK**.

    The network ACL rule list page is displayed and the target rules are selected automatically.
11. Click **Delete** above the network ACL rule list.

    A confirmation dialog box is displayed.
12. Confirm the information and click **OK**.

6.3.5 Managing Subnets Associated with a Network ACL

6.3.5.1 Associating Subnets with a Network ACL

Scenarios

You can associate a subnet with a network ACL. If it is enabled, it controls traffic in and out of the subnet.

Associating subnets with a network ACL may affect how and where traffic is directed. Be careful with this operation as it may interrupt services.

Constraints

- You can associate a network ACL with multiple subnets. However, a subnet can only be associated with one network ACL at a time.
- After a network ACL is associated with a subnet, the default rules deny all traffic to and from the subnet until you add custom rules to allow traffic. For details, see **Adding a Network ACL Rule**.

Procedure

1. Log in to the management console.
2. Click in the upper left corner and select the desired region and project.
3. Click in the upper left corner and choose **Networking** > **Virtual Private Cloud**.

   The **Virtual Private Cloud** page is displayed.
4. Associate a subnet with a network ACL using either of the following methods:
   - Method 1
     i. In the navigation pane on the left, click **Subnets**.

        The **Subnets** page is displayed.

     ii. In the subnet list, locate the row that contains the subnet and click **Associate** under the **Network ACL** column.

        The **Associate Network ACL** page is displayed.

     iii. Select a network ACL from the drop-down list.

        If there is no network ACL, click in the drop-down list to create one.

     iv. Click **OK**.

        The subnet list is displayed. You can view the associated network ACL of the subnet.
   - Method 2
     i. In the navigation pane on the left, choose **Access Control** > **Network ACLs**.

        The network ACL list is displayed.

ii. In the network ACL list, locate the target network ACL and click **Associate Subnet** in the **Operation** column.

The **Associated Subnets** tab is displayed.

iii. On the **Associated Subnets** tab, click **Associate**.

The **Associate Subnet** dialog box is displayed.

iv. In the **Associate Subnet** dialog box, select the subnet from the subnet list and click **OK**.

In the associated subnet list, you can view all subnets associated with the network ACL.

NOTE

A subnet with a network ACL associated will not be displayed in the subnet list of the **Associate Subnet** dialog box for you to select. If you want to associate such a subnet with another network ACL, you must first disassociate the subnet from the original network ACL.

6.3.5.2 Disassociating Subnets from a Network ACL

Scenarios

You can disassociate a subnet from a network ACL based on your network requirements.

Procedure

1. Log in to the management console.
2. Click in the upper left corner and select the desired region and project.
3. Click in the upper left corner and choose **Networking** > **Virtual Private Cloud**.

   The **Virtual Private Cloud** page is displayed.
4. Disassociate a subnet from a Networking using the following methods:
   - Method 1
     i. In the navigation pane on the left, click **Subnets**.

     The **Subnets** page is displayed.

     ii. In the subnet list, locate the target subnet and click its name.

     The subnet details page is displayed.

     iii. In the upper right corner of the subnet details page, click **Disassociate** next to the network ACL.

     A confirmation dialog box is displayed.

     iv. Confirm the information and click **OK**.

     On the subnet details page, you can see that no network ACL is associated with the subnet.
   - Method 2
     i. In the navigation pane on the left, click **Subnets**.

     The **Subnets** page is displayed.

ii. In the subnet list, locate the target subnet and click the name of the network ACL under the **Network ACL** column.

The network ACL details page is displayed.

iii. Click the **Associated Subnets** tab, select one or more subnets, and click **Disassociate** in the **Operation** column.

A confirmation dialog box is displayed.

iv. Click **OK** in the displayed dialog box.

On the **Associated Subnets** tab, you cannot see the disassociated subnets in the subnet list.

- Method 3

i. In the navigation pane on the left, choose **Access Control** > **Network ACLs**.

The network ACL list is displayed.

ii. In the network ACL list, locate the target network ACL and click **Associate Subnet** in the **Operation** column.

The **Associated Subnets** tab is displayed.

iii. Select one or more subnets and click **Disassociate**.

A confirmation dialog box is displayed.

iv. Click **OK** in the displayed dialog box.

On the **Associated Subnets** tab, you cannot see the disassociated subnets in the subnet list.

7 IP Address Group

7.1 IP Address Group Overview

What Is an IP Address Group?

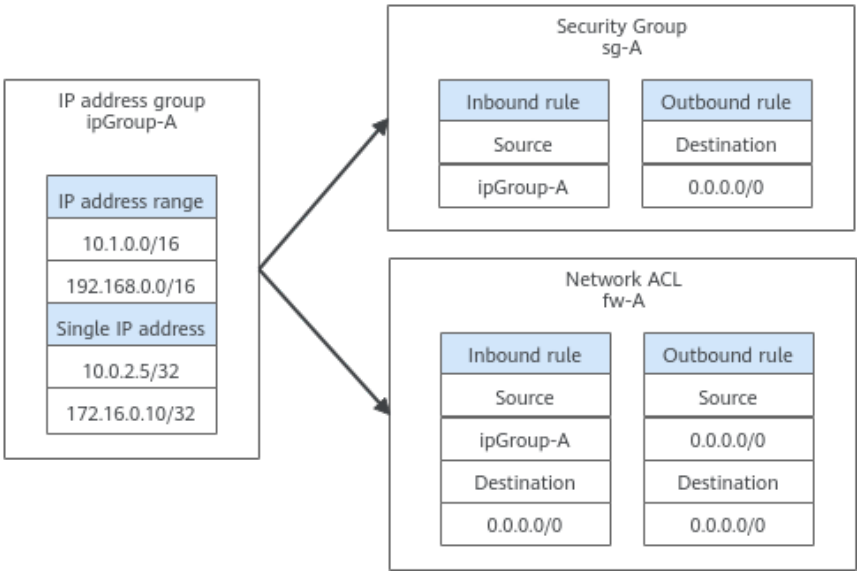
An IP address group is a collection of IP addresses. It can be associated with security groups and network ACLs to simplify IP address configuration and management.

You can add IP address ranges and IP addresses that need to be managed in a unified manner to an IP address group. An IP address group can work together with different cloud resources. **Table 7-1** lists the resources that can be associated with an IP address group.

Table 7-1 Resources that can be associated with an IP address group

Resource	Description	Example
Security group	The **Source** or **Destination** of a security group rule can be set to **IP address group**.	As shown in **Figure 7-1**, the inbound rule of security group **sg-A** uses IP address group **ipGroup-A** as the source.
Network ACL	The **Source** or **Destination** of a network ACL is set to **IP address group**.	As shown in **Figure 7-1**, the inbound rule of network ACL **fw-A** uses IP address group **ipGroup-A** as the source.

Figure 7-1 Using IP address group



Notes

If you have multiple IP addresses with the same security requirements, you can add them to an IP address group and select this IP address group when you configure a rule, to help you manage them in an easier way. When an IP address changes, you only need to change the IP address in the IP address group. Then, the rules in the IP address group change accordingly. You do not need to modify the rules in the security group one by one. This simplifies security group management and improves efficiency. For details, see **Using IP Address Groups to Reduce the Number of Security Group Rules**.

Constraints

- Security group rules that are associated with an IP address group do not take effect for certain ECSs.
  - General computing (S1, C1, and C2 ECSs)
  - Memory-optimized (M1 ECSs)
  - High-performance computing (H1 ECSs)
  - Disk-intensive (D1 ECSs)
  - GPU-accelerated (G1 and G2 ECSs)
  - Large-memory (E1, E2, and ET2 ECSs)
- If a network ACL rule uses an IP address group:
  - Either the source or the destination of an inbound rule can use the IP address group.
  - Either the source or the destination of an outbound rule can use the IP address group.

  For example, if the source of an inbound rule network ACL is set to an IP address group, the rule destination can only be an IP address.

7.2 Managing an IP Address Group

7.2.1 Creating an IP Address Group

Scenarios

This section describes how to create an IP address group. An IP address group is a collection of IP addresses that can be associated with security groups and network ACLs to simplify IP address configuration and management.

Procedure

1. Go to the **Create IP Address Group** page.
2. Configure the parameters as prompted.
   For details, see **Table 7-2**.

Table 7-2 Parameters for creating an IP address group

Parameter	Description	Example Value
Region	Mandatory The region where the IP address group belongs. Select the region nearest to you to ensure the lowest latency possible. An IP address group can be associated only with resources in the same region.	Region A
Name	Mandatory Enter the name of the IP address group. The name: • Can contain 1 to 64 characters. • Can contain letters, digits, underscores (_), hyphens (-), and periods (.). You can customize the name of an IP address group that is uniquely identified by its ID.	ipGroup-A

Parameter	Description	Example Value
Max. IP Addresses	Mandatory Set the number of IP addresses that can be added to an IP address group. By default, the system displays the maximum number of IP addresses that can be added to an IP address group. You can change number as required. If you want to increase the maximum number of IP addresses in a group, **submit a service ticket**.	20
IP Address Version	Mandatory Select the type of IP addresses that can be added to an IP address group. • **IPv4** • **IPv6**	IPv4
IP Addresses (Optional)	Optional Enter an IP address or IP address range on each line, and press **Enter**. The format is "IP address | Description". Description is optional, can contain 0 to 255 characters, and cannot contain angle brackets (< or >). You can enter: • An IPv4 address range, for example, 192.168.0.0/16 or 192.168.0.0/16 | ECS01 • A single IPv4 address, for example, 192.168.10.10 or 192.168.10.10 | ECS01 • An IPv6 address range, for example, 2001:db8:a583:6e::/64 or 2001:db8:a583:6e::/64 | ECS01 • A single IPv6 address, for example, 2001:db8:a583:6e::5c or 2001:db8:a583:6e::5c | ECS01	• Without description: 192.168.0.0/16 • With description: 192.168.0.0/16 | ECS01

Parameter	Description	Example Value
Enterprise Project	Mandatory When creating an IP address group, you can add the group to an enabled enterprise project. An enterprise project facilitates project-level management and grouping of cloud resources and users. The default project is **default**. For details about creating and managing enterprise projects, see the *Enterprise Management User Guide*.	default
Description (Optional)	Optional Enter the description of the IP address group in the text box as required.	-

3. Click **Create Now**.

   The IP address group list is displayed. The status of the created IP address group is **Normal**.

NOTICE

An IP address group takes effect only after it is associated with corresponding resources. For details, see **Associating an IP Address Group with Resources**.

7.2.2 Associating an IP Address Group with Resources

Scenarios

This section describes how to associate an IP address group with a resource.

An IP address group can be associated with security groups and network ACLs.

Prerequisites

- You have created an IP address group. For details, see **Creating an IP Address Group**.
- You have added IP addresses to the IP address group. For details, see **Adding IP Addresses to an IP Address Group**.

Procedure

You need to associate an IP address group with resources. For details, see **Table 7-3**.

Table 7-3 Associating an IP address group with resources

Resource	Description	Reference
Security group	The **Source** or **Destination** of a security group rule can be set to **IP address group**.	**Adding a Security Group Rule** • Inbound rule: Set **Source** to an IP address group. • Outbound rule: Set **Destination** to an IP address group.
Network ACL	The **Source** or **Destination** of a network ACL is set to **IP address group**.	**Adding a Network ACL Rule** • Inbound rule: Set **Source** or **Destination** to an IP address group. Either the source or the destination can use the IP address group. • Outbound rule: Set **Source** or **Destination** to an IP address group. Either the source or the destination can use the IP address group.

7.2.3 Disassociating an IP Address Group from Resources

Scenarios

This section describes how to disassociate an IP address group from a resource.

An IP address group can be associated with security groups and network ACLs.

Notes and Constraints

Disassociating an IP address group from resources will make the rules of the resources invalid, and this action cannot be undone.

Procedure

1. Go to the **IP address group list page**.
2. In the IP address group list, locate the target IP address group and click the resource name in the **Associated Resources** column.

   The **Associated Resources** page is displayed.
3. In the **Associated Resources** list, click the corresponding resource name.

   The resource summary page is displayed. You can refer to **Table 7-4** to disassociate the IP address group from resources.

Table 7-4 Disassociating an IP address group from resources

Resource	Description	Reference
Security group	Modify or delete inbound or outbound rules associated with the IP address group.	• **Modifying a Security Group Rule** – Inbound rule: Change the value of **Source**. – Outbound rule: Change the value of **Destination**. • **Deleting One or More Security Group Rules**
network ACL	Modify or delete inbound or outbound rules associated with the IP address group.	• **Modifying a Network ACL Rule** – Inbound rule: Change the value of **Source** or **Destination**. – Outbound rule: Change the value of **Source** or **Destination**. • **Deleting One or More Network ACL Rules**

7.2.4 Modifying an IP Address Group

Scenarios

This section describes how to modify basic information about an IP address group, including:

- Name
- Description

Procedure

1. Go to the **IP address group list page**.
2. In the IP address group list, click the name of the target IP address group.
   The basic information page of the IP address group is displayed.
3. On the **Basic Information** tab page of the IP address group, click  on the right of the target parameter and modify the parameter as prompted.
   For details, see **Table 7-5**.

Table 7-5 IP address group parameters

Parameter	Description	Example Value
Name	Mandatory Enter the name of the IP address group. The name: • Can contain 1 to 64 characters. • Can contain letters, digits, underscores (_), hyphens (-), and periods (.). You can customize the name of an IP address group that is uniquely identified by its ID.	ipGroup-A
Max. IP Addresses	Mandatory Set the number of IP addresses that can be added to an IP address group. By default, the system displays the maximum number of IP addresses that can be added to an IP address group. You can change number as required. If you want to increase the maximum number of IP addresses in a group, **submit a service ticket**.	20
Description	Optional Enter the description of the IP address group in the text box as required.	-

4. Click .

7.2.5 Exporting IP Address Group Details

Scenarios

This section describes how to export details about IP address groups, including:

- Name, ID, and creation time
- Added IP addresses
- Associated resources

Procedure

1. Go to the **IP address group list page**.
2. In the upper left corner above the IP address group list, click **Export**.

- **Export selected data to an XLSX file**: Select one or more IP address groups and export information about the selected IP address groups.
- **Export all data to an XLSX file**: Export information about all the IP address groups in the current region.

The system will automatically export information about the IP address groups as an Excel file to a local directory.

7.2.6 Viewing the Details of an IP Address Group

Scenarios

This section describes how to view information about an IP address group, including:

- Name, ID, and creation time
- Added IP addresses
- Associated resources

Procedure

1. Go to the **IP address group list page**.
2. In the IP address group list, click the hyperlink of the IP address group name.

   The basic information page of the IP address group is displayed.
3. Click different tabs to view the required information.
   a. On the **Basic Information** tab page, view the basic information and IP addresses added to the IP address group.
   b. On the **Associated Resources** tab page, view the resources associated with the IP address group.

7.2.7 Managing IP Address Group Tags

Scenarios

Tags help you identify, classify, and search for IP address groups. You can perform the following operations to manage the tags of an IP address group:

- Add an IP address group tag.
- Modify an IP address group tag.
- Delete an IP address group tag.

For details about IP address group tag requirements, see **Table 7-6**.

Table 7-6 IP address group tag key and value requirements

Parameter	Requirements	Example Value
Tag key	• For each resource, each tag key must be unique, and each tag key can only have one tag value. • Cannot be left blank. • Can contain a maximum of 128 characters. • Can contain letters in any language, digits, spaces, underscores (_), periods (.), colons (:), equal signs (=), plus signs (+), minus signs (-), and at signs (@). • Cannot start with _sys_ or a space or end with a space.	test
Tag value	• Can be left blank. • Can contain a maximum of 255 characters. • Can contain letters in any language, digits, spaces, underscores (_), periods (.), colons (:), slashes (/), equal signs (=), plus signs (+), minus signs (-), and at signs (@). • Cannot start or end with a space.	01

Notes and Constraints

Each cloud resource can have a maximum of 20 tags.

Procedure

1. Log in to the management console.
2. Click in the upper left corner and select the desired region and project.
3. Click in the upper left corner and choose **Networking** > **Virtual Private Cloud**.

   The **Virtual Private Cloud** page is displayed.
4. Go to the **IP address group list page**.
5. On the **Tags** tab, click **Edit Tag** in the upper left corner above the tag list.

   The **Edit Tag** dialog box is displayed.
6. Perform the following operations on the tag as required:
   - Adding a tag: Click +, enter a tag key and value, and click **OK**.
   - Modifying a tag: Click × next to the target tag key or value, delete the original value, enter a new value, and click **OK**.

- Deleting a tag: Click **Delete** next to the target tag and click **OK**.

7.2.8 Deleting an IP Address Group

Scenarios

This section describes how to delete an IP address group.

Notes and Constraints

If an IP address group has been associated with a resource, deleting the IP address group will delete the rules that use the IP address group for the associated resource. This interrupts network connectivity.

Procedure

1. Go to the **IP address group list page**.
2. In the IP address group list, delete IP address groups.
   - Delete a single IP address group:
     i. In the IP address list, locate the row that contains the IP address group and click **Delete** in the **Operation** column.
        A confirmation dialog box is displayed.
     ii. Confirm the information and click **OK**.
   - Delete IP address groups in batches.
     i. In the IP address list, select the IP address groups to be deleted.
     ii. Click the **Delete** button located above the IP address group list.
        A confirmation dialog box is displayed.
     iii. Confirm the information and click **OK**.
        If a message indicating that IP address groups with associated resources cannot be deleted is displayed, go to the resource details page and **disassociate the IP address group from the resources first**.

7.3 Managing IP Addresses in an IP Address Group

7.3.1 Adding IP Addresses to an IP Address Group

Scenarios

This section describes how to add IP addresses to an IP address group.

Notes and Constraints

If an IP address group has resources associated, adding IP addresses to the group may affect your network communications.

If an IP address group is associated with security groups and network ACLs, the rules associated with the IP address groups will change.

Procedure

1. Go to the **IP address group list page**.
2. In the IP address group list, click the name of the target IP address group.
   The basic information page of the IP address group is displayed.
3. In the left corner above the IP address list, click **Add**.
   The **Add IP Address** dialog box is displayed.
4. Add IP addresses to the IP address group as prompted.
   - Method 1
     i. Enter IP addresses in the **IP Addresses** box. For details, see **Table 7-7**.

Table 7-7 Parameters for adding IP addresses

Parameter	Description	Example Value
Name	The name of the IP address group.	ipGroup-A
IP Address Version	IP address version supported by an IP address group. You can select an IP address version when you create an IP address group. IP address version cannot be modified after the IP address group is created. The supported versions are as follows: • IPv4 • **IPv6**	IPv4

Parameter	Description	Example Value
IP Addresses	Mandatory Enter an IP address or IP address range on each line, and press **Enter**. The format is "IP address | Description". Description is optional, can contain 0 to 255 characters, and cannot contain angle brackets (< or >). You can enter: • An IPv4 address range, for example, 192.168.0.0/16 or 192.168.0.0/16 | ECS01 • A single IPv4 address, for example, 192.168.10.10 or 192.168.10.10 | ECS01 • An IPv6 address range, for example, 2001:db8:a583:6e::/64 or 2001:db8:a583:6e::/64 | ECS01 • A single IPv6 address, for example, 2001:db8:a583:6e::5c or 2001:db8:a583:6e::5c | ECS01	• Without description: 192.168.0.0/16 • With description: 192.168.0.0/16 | ECS01

ii. Click **OK**.

In the IP address list, you can view the newly added IP addresses.

– Method 2

Click **Batch Import** in the lower part of the **IP Addresses** box. On the displayed **Batch Import IP Addresses** dialog box, import IP addresses by referring to **Importing IP Addresses to an IP Address Group in Batches**.

7.3.2 Modifying IP Addresses in an IP Address Group

Scenarios

This section describes how to modify IP addresses, IP address ranges, and their descriptions in an IP address group.

Notes and Constraints

If an IP address group has resources associated, modifying IP addresses in an IP address group may affect your network communications.

If an IP address group is associated with security groups and network ACLs, the rules associated with the IP address groups will change.

Procedure

1. Go to the **IP address group list page**.
2. In the IP address group list, click the name of the target IP address group.
   The basic information page of the IP address group is displayed.
3. In the left corner above the IP address list, click **Modify**.
   The **Modify IP Address** dialog box is displayed.
4. Modify the information as prompted.
   For details, see **Table 7-8**.

Table 7-8 Parameters for modifying IP addresses

Parameter	Description	Example Value
Name	The name of the IP address group.	ipGroup-A
Max. IP Addresses	Mandatory Set the number of IP addresses that can be added to an IP address group. By default, the system displays the maximum number of IP addresses that can be added to an IP address group. You can change number as required. If you want to increase the maximum number of IP addresses in a group, **submit a service ticket**.	20
IP Address Version	IP address version supported by an IP address group. You can select an IP address version when you create an IP address group. IP address version cannot be modified after the IP address group is created. The supported versions are as follows: • IPv4 • IPv6	IPv4

Parameter	Description	Example Value
IP Addresses	You can modify existing IP addresses, IP address ranges, and their descriptions in an IP address group. The format is "IP address | Description". Description is optional, can contain 0 to 255 characters, and cannot contain angle brackets (< or >). You can enter: • An IPv4 address range, for example, 192.168.0.0/16 or 192.168.0.0/16 | ECS01 • A single IPv4 address, for example, 192.168.10.10 or 192.168.10.10 | ECS01 • An IPv6 address range, for example, 2001:db8:a583:6e::/64 or 2001:db8:a583:6e::/64 | ECS01 • A single IPv6 address, for example, 2001:db8:a583:6e::5c or 2001:db8:a583:6e::5c | ECS01	• Without description: 192.168.0.0/16 • With description: 192.168.0.0/16 | ECS01

5. Click **OK**.

   The IP address list is displayed and you can view that the IP address was modified.

7.3.3 Importing IP Addresses to an IP Address Group in Batches

Scenarios

You can enter IP addresses, IP address ranges, and their descriptions in an Excel file and import the file to an IP address group. This allows you to quickly add multiple IP addresses.

Notes and Constraints

- The number of IP addresses that can be imported is limited. You can check the limited quota on the console.
- Duplicate IP addresses will not be imported, for example:
  - Both the IP address ranges and their descriptions are the same.
  - The IP address ranges are the same but their descriptions are different.

Procedure

1. Go to the **IP address group list page**.

2. In the IP address group list, click the name of the target IP address group.
   The basic information page of the IP address group is displayed.
3. In the left corner above the IP address list, click **Import**.
   The **Batch Import IP Addresses** dialog box is displayed.
4. Click **Download Template** to download the Excel template.
5. In the Excel file, enter IP addresses, IP address ranges, and their descriptions, and save the file.
   For details about parameter in the Excel file, see **Table 7-9**.

   Table 7-9 Parameters for importing IP addresses

Parameter	Description	Example Value
IP Addresses	Mandatory In the **IP Addresses** column, enter an IP address range or a single IP address on a separate line. You can enter: • An IPv4 address range, for example, 192.168.0.0/16 • A single IPv4 address, for example, 192.168.10.10 • An IPv6 address range, for example, 2001:db8:a583:6e::/64 • A single IPv6 address, for example, 2001:db8:a583:6e::5c	192.168.0.0/16
Description	Optional In the **Description** column, enter the description of the IP address or IP address range. Description can contain 0 to 255 characters, and cannot contain angle brackets (<>).	ECS01

6. In the **Batch Import IP Addresses** dialog box, click **Select File**, select the Excel file, and click **Import**.
   After the import is complete, you can view the newly imported IP addresses, IP address ranges, and their descriptions.

7.3.4 Deleting IP Addresses from an IP Address Group

Scenarios

This section describes how to delete IP addresses from an IP address group.

Notes and Constraints

If an IP address group has resources associated, deleting IP addresses from the group may affect your network communications.

If an IP address group is associated with security groups and network ACLs, the rules associated with the IP address groups will change.

Procedure

1. Go to the **IP address group list page**.
2. In the IP address group list, click the name of the target IP address group.

   The basic information page of the IP address group is displayed.
3. Delete IP addresses:
   - Delete a single IP address.
     i. In the IP address list, locate the target IP address and click **Delete** in the **Operation** column.

        A confirmation dialog box is displayed.
     ii. Confirm the information and click **OK**.
   - Delete IP addresses in batches.
     i. In the IP address list, select the IP addresses to be deleted.
     ii. Click the **Delete** button above the IP address list.

        A confirmation dialog box is displayed.
     iii. Confirm the information and click **OK**.

7.4 IP Address Group Configuration Examples

7.4.1 Using IP Address Groups to Reduce the Number of Security Group Rules

Scenarios

An IP address group is a collection of one or more IP addresses. You can use IP address groups when configuring security group rules. If you change the IP addresses in an IP address group, the security group rules are changed accordingly. You do not need to modify the security group rules one by one.

Finance and securities enterprises have high security requirements when planning cloud networks. Access to instances is often controlled based on IP addresses. To simplify security group rule configuration and control access based on IP addresses, you can use IP address groups to manage IP address ranges and IP addresses with the same security requirements. For more information about IP address groups, see **IP Address Group Overview**.

Suppose your enterprise has an online office system deployed on the cloud. To provide services for different departments, you associate office servers with different security groups based on security levels. These servers are accessed from a large number of IP addresses that may change from time to time.

- If IP address groups are not used, you need to configure multiple rules to control access from different sources. Once the IP addresses change, you need to adjust the rules in each security group one by one. The management workload increases with the number of security groups and rules.
- If IP address groups are used, you can add the IP addresses with the same security requirements to an IP address group and add rules with source set to this IP address group. When an IP address changes, you only need to change it in the IP address group. Then, the security group rules using the IP address group change accordingly. You do not need to modify the security group rules one by one. This simplifies security group management and improves efficiency.

Solution Architecture

In this practice, the instances are associated with three security groups based on different security requirements. In addition, these instances need to be accessed by specific IP addresses over SSH port 22. To simplify management, you can use IP address groups.

1. Create an IP address group and add IP addresses that need to access the instances.
2. Add inbound rules to allow traffic from the IP address group to the instances in the three security groups.

   Table 7-10 Inbound rules

Direction	Action	Type	Protocol & Port	Source
Inbound	Allow	IPv4	TCP:22	IP address group

3. Change the IP addresses in the IP address group if any IP addresses change. Then, the rules using the IP address group change accordingly.

Constraints

Security group rules using IP address groups do not take effect for the following instances:

- General computing (S1, C1, and C2 ECSs)
- Memory-optimized (M1 ECSs)
- High-performance computing (H1 ECSs)
- Disk-intensive (D1 ECSs)
- GPU-accelerated (G1 and G2 ECSs)
- Large-memory (E1, E2, and ET2 ECSs)

Resource Planning

In this practice, the IP address group and security groups must be in the same region. For details, see **Table 7-11**. The following resource details are only examples. You can modify them as required.

Table 7-11 Resource planning

Resource	Quantity	Description
IP address group	1	Create an IP address group and add IP addresses that need to access the instances. • **Name**: **ipGroup-A** • **Max. IP Addresses**: Set it as required. In this practice, **20** is used. • **IP Address Version**: Set it as required. In this practice, **IPv4** is used. • **IP Addresses**: – 11.xx.xx.64/32 – 116.xx.xx.252/30 – 113.xx.xx.0/25 – 183.xx.xx.208/28
Security group	3	Add inbound rules to allow traffic from **ipGroup-A** to the instances in the three security groups, as shown in **Table 7-12**.

Table 7-12 Inbound rules

Direction	Action	Type	Protocol & Port	Source
Inbound	Allow	IPv4	TCP:22	ipGroup-A

Procedure

Step 1 Create IP address group **ipGroup-A** and add IP addresses that need to access the instances.

For details, see **Creating an IP Address Group**.

Step 2 Add inbound rules to allow traffic from **ipGroup-A** to the instances in the three security groups.

For details, see **Adding a Security Group Rule**.

After the rules are added, traffic from 11.xx.xx.64/32, 116.xx.xx.252/30, 113.xx.xx.0/25, and 183.xx.xx.208/28 are allowed to the Linux ECSs over SSH port 22.

Step 3 Change IP addresses in the IP address group.

After security group rules are added, you can add IP addresses to **ipGroup-A**. For example, you can add 117.xx.xx.0/25 to **ipGroup-A**, and the security groups rule is applied automatically, allowing traffic from 117.xx.xx.0/25 over SSH port 22.

For details, see **Managing IP Addresses in an IP Address Group**.

----End

8 VPC Peering Connection

8.1 VPC Peering Connection Overview

What Is a VPC Peering Connection?

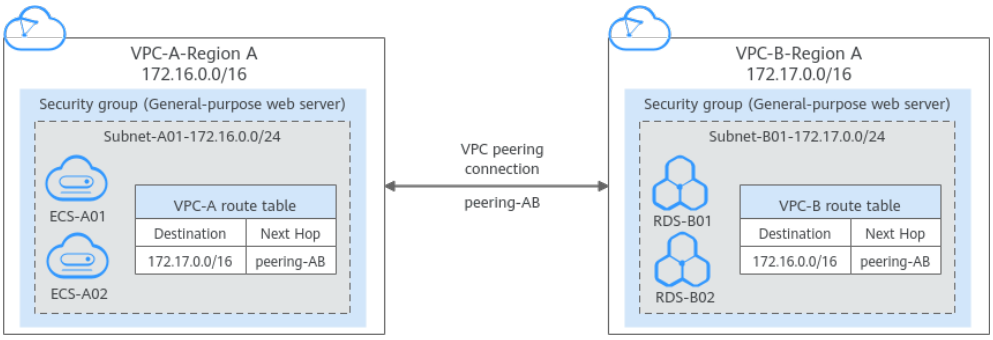
A VPC peering connection enables two VPCs in the same region to communicate using private IP addresses. The VPCs to be connected can be from the same account or different accounts.

- If you want to connect VPCs in different regions, use **Cloud Connect**.
- You can use VPC peering connections to build different networks. For details, see **VPC Peering Connection Usage Examples**.

Figure 8-1 shows an application scenario of VPC peering connections.

- There are two VPCs (VPC-A and VPC-B) in region A that are not connected.
- Service servers (ECS-A01 and ECS-A02) are in VPC-A, and database servers (RDS-B01 and RDS-B02) are in VPC-B. The service servers and database servers cannot communicate with each other.
- You need to create a VPC peering connection (peering-AB) between VPC-A and VPC-B so the service servers and database servers can communicate with each other.

Figure 8-1 Two VPCs connected by a VPC peering connection



NOTICE

Currently, VPC peering connections are free.

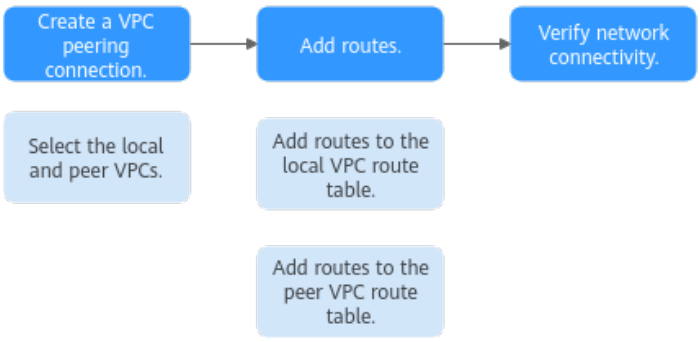
VPC Peering Connection Creation Process

A VPC peering connection can only connect VPCs in the same region.

- If two VPCs are in the same account, the process of creating a VPC peering connection is shown in **Figure 8-2**.

  For details about how to create a VPC peering connection, see **Creating a VPC Peering Connection to Connect Two VPCs in the Same Account**.

  Figure 8-2 Process of creating a VPC peering connection between VPCs in the same account

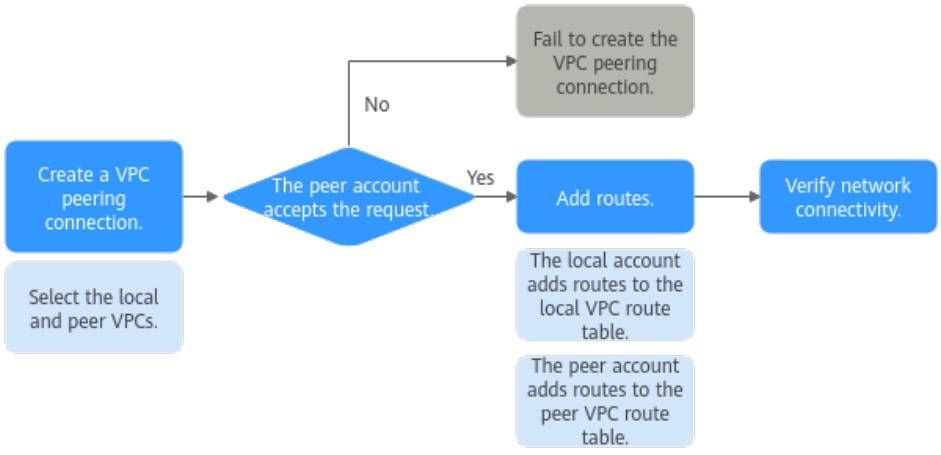
  

- If two VPCs are in different accounts, the process of creating a VPC peering connection is shown in **Figure 8-3**.

  For details about how to create a VPC peering connection, see **Creating a VPC Peering Connection to Connect Two VPCs in Different Accounts**.

  If account A initiates a request to create a VPC peering connection with a VPC in account B, the VPC peering connection takes effect only after account B accepts the request.

  Figure 8-3 Process of creating a VPC peering connection between VPCs in different accounts

Notes and Constraints

- A VPC peering connection can only connect VPCs in the same region.
  - A VPC peering connection can enable a VPC created on the Huawei Cloud Chinese Mainland website to connect to one created on the Huawei Cloud International website, but the VPCs must be in the same region. For example, if the VPC created on the Chinese Mainland website is in the CN-Hong Kong region, then the VPC created on the International website must also be in the CN-Hong Kong region.
  - If you want to connect VPCs in different regions, you can use **Cloud Connect**.
  - If you only need a few ECSs in different regions to communicate with each other, you can **assign and bind EIPs to the ECSs**.
- If the local and peer VPCs have overlapping CIDR blocks, the VPC peering connection may not be usable.

  In this case, you can configure the network by referring to **VPC Peering Connection Usage Examples**.
- By default, if VPC A is peered with VPC B that has EIPs, VPC A cannot use EIPs in VPC B to access the Internet. To enable this, you can use the NAT Gateway service or configure an SNAT server. For details, see **Enabling Internet Connectivity for an ECS Without an EIP**.

8.2 VPC Peering Connection Usage

8.2.1 VPC Peering Connection Usage Examples

A VPC peering connection is a networking connection between two VPCs in the same region and enables them to communicate. **Table 8-1** lists different scenarios of using VPC peering connections.

Table 8-1 VPC peering connection usage examples

Location	CIDR Block	Description	Example
VPCs in the same region	• VPC CIDR blocks do not overlap. • Subnet CIDR blocks of VPCs do not overlap.	You can create VPC peering connections to connect entire CIDR blocks of VPCs. Then, all resources in the VPCs can communicate with each other.	**Using a VPC Peering Connection to Connect Two VPCs**

Location	CIDR Block	Description	Example
VPCs in the same region	• VPC CIDR blocks overlap. • Some subnet CIDR blocks overlap.	You can create VPC peering connections to connect specific subnets or ECSs from different VPCs. • To connect specific subnets from two VPCs, the subnet CIDR blocks cannot overlap. • To connect specific ECSs from two VPCs, each ECS must have a unique private IP address.	**Using a VPC Peering Connection to Connect Subnets in Two VPCs** **Using a VPC Peering Connection to Connect ECSs in Two VPCs**
VPCs in the same region	• VPC CIDR blocks overlap. • All subnet CIDR blocks overlap.	VPC peering connections are not usable.	**Unsupported VPC Peering Configurations**

NOTICE

A VPC peering connection can only connect VPCs in the same region. If your VPCs are in different regions, use **Cloud Connect**.

Alternatively, you can use enterprise routers to connect VPCs in the same region. **Enterprise Router** is more suitable for complex networking that needs to connect multiple VPCs. With enterprise routers, you do not have to create a large number of VPC peering connections or add too many routes. This makes your network topology simpler and more scalable.

All route tables in a VPC can have a maximum of 1,000 routes. If you want to create VPC peering connections to connect multiple VPCs, consider this restriction when planning the networking.

8.2.2 Using a VPC Peering Connection to Connect Two VPCs

You can configure a VPC peering connection and set the destination of the routes added to VPC route tables to the peer VPC CIDR block. In this way, all resources in the two VPCs are connected. **Table 8-2** shows example scenarios.

Table 8-2 Scenario description

Scenario	Scenario Description	IP Address Version	Example
Two VPCs peered together	You have two VPCs that require full access to each other's resources. For example, your company has VPC-A for the human resource department, and VPC-B for the finance department. The two departments require full access to each other's resources.	IPv4	**Two VPCs Peered Together (IPv4)**
		IPv6	**Two VPCs Peered Together (IPv6)**
Multiple VPCs peered together	You have multiple VPCs that require access to each other's resources. For example, your company has VPC-A for the human resource department, VPC-B for the finance department, and VPC-C for the marketing department. These departments require full access to each other's resources.	IPv4	**Multiple VPCs Peered Together (IPv4)**
		IPv4	**Multiple VPCs Peered Together Through Transitive Peering Connections (IPv4)**
		IPv6	**Multiple VPCs Peered Together (IPv6)**
One central VPC peered with two VPCs	You have a central VPC that requires access to two peer VPCs, and similarly, the peer VPCs require access to the central VPC. However, the two peer VPCs need to be isolated from each other. For example, public services (such as databases) are deployed on VPC-A. Both VPC-B and VPC-C need to access the databases, but they do not need to access each other.	IPv4	**One Central VPC Peered with Two VPCs (IPv4)**
		IPv6	**One Central VPC Peered with Two VPCs (IPv6)**
One central VPC with primary and secondary CIDR blocks peered with two VPCs	You have a central VPC that has both primary and secondary CIDR blocks. The central VPC needs to communicate with two peer VPCs, but the peer VPCs need to be isolated from each other.	IPv4	**One Central VPC with Primary and Secondary CIDR Blocks Peered with Two VPCs (IPv4)**

Scenario	Scenario Description	IP Address Version	Example
One central VPC peered with multiple VPCs	You have a central VPC that requires access to the multiple peer VPCs, and similarly, the peer VPCs require access to the central VPC. However, the peer VPCs need to be isolated from each other. For example, public services (such as databases) are deployed on your central VPC-A. VPC-B, VPC-C, VPC-D, VPC-E, VPC-F, and VPC-G need to access the databases, but these VPCs do not need to access each other.	IPv4	**One Central VPC Peered with Multiple VPCs (IPv4)**
		IPv6	**One Central VPC Peered with Multiple VPCs (IPv6)**

Notes and Constraints

If you create a VPC peering connection that connects entire CIDR blocks of two VPCs, the VPC CIDR blocks cannot overlap. Otherwise, the VPC peering connection does not take effect. For details, see **Invalid VPC Peering for Overlapping VPC CIDR Blocks**.

Even if you intend to use the VPC peering connection for IPv6 communication only, you cannot create a VPC peering connection if the VPCs have matching or overlapping IPv4 CIDR blocks. In all examples in this section, the IPv4 CIDR blocks of any VPCs connected by a VPC peering connection do not overlap.

Two VPCs Peered Together (IPv4)

Create Peering-AB between VPC-A and VPC-B. The CIDR blocks of VPC-A and VPC-B do not overlap.

- For details about resource planning, see **Table 8-3**.
- For details about VPC peering relationships, see **Table 8-4**.

Figure 8-4 Networking diagram (IPv4)

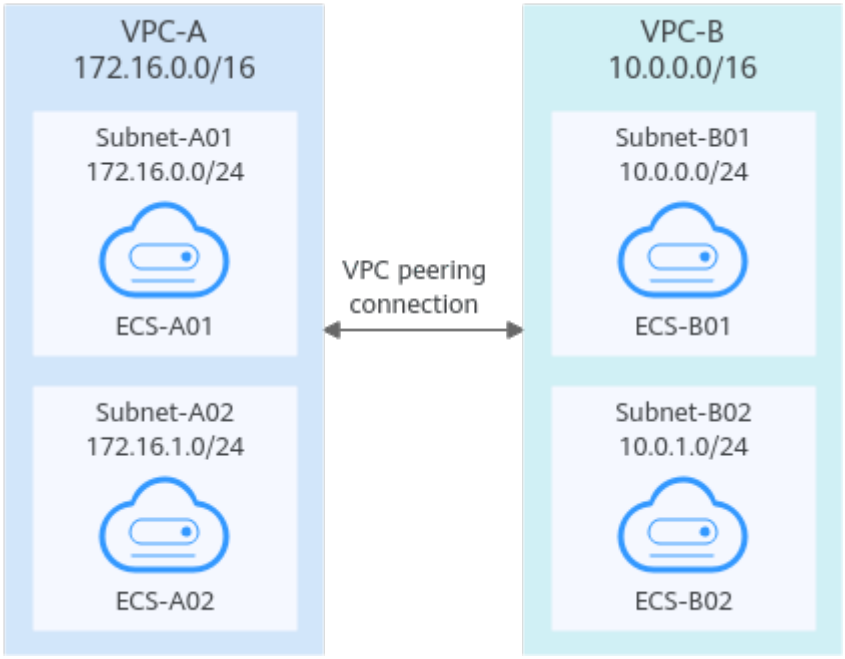


Table 8-3 Resource planning details (IPv4)

VPC Name	VPC CIDR Block	Subnet Name	Subnet CIDR Block	Subnet Route Table	ECS Name	Security Group	Private IP Address
VPC-A	172.16.0.0/16	Subnet-A01	172.16.0.0/24	rtb-VPC-A	ECS-A01	sg-web: general-purpose web server	172.16.0.111
		Subnet-A02	172.16.1.0/24	rtb-VPC-A	ECS-A02		172.16.1.91
VPC-B	10.0.0.0/16	Subnet-B01	10.0.0.0/24	rtb-VPC-B	ECS-B01		10.0.0.139
		Subnet-B02	10.0.1.0/24	rtb-VPC-B	ECS-B02		10.0.1.167

Table 8-4 Peering relationships (IPv4)

Peering Relationship	Peering Connection Name	Local VPC	Peer VPC
VPC-A is peered with VPC-B.	Peering-AB	VPC-A	VPC-B

After the VPC peering connection is created, add the following routes to the route tables of the local and peer VPCs:

Table 8-5 VPC route tables (IPv4)

Route Table	Destination	Next Hop	Route Type	Description
rtb-VPC-A	172.16.0.0/24	Local	System	Local routes are automatically added for communications within a VPC.
	172.16.1.0/24	Local	System	
	10.0.0.0/16 (VPC-B)	Peering-AB	Custom	Add a route with the CIDR block of VPC-B as the destination and Peering-AB as the next hop.
rtb-VPC-B	10.0.0.0/24	Local	System	Local routes are automatically added for communications within a VPC.
	10.0.1.0/24	Local	System	
	172.16.0.0/16 (VPC-A)	Peering-AB	Custom	Add a route with the CIDR block of VPC-A as the destination and Peering-AB as the next hop.

NOTE

If the destination of a route in a route table of a VPC is set to the CIDR block of a peer VPC and the CIDR blocks of the two VPCs do not overlap, the VPCs can have full access to each other's resources.

Two VPCs Peered Together (IPv6)

Create Peering-AB between VPC-A and VPC-B. The subnets of VPC-A and VPC-B have both IPv4 and IPv6 CIDR blocks and their IPv4 CIDR blocks do not overlap.

- For details about resource planning, see **Table 8-6**.
- For details about VPC peering relationships, see **Table 8-7**.

Figure 8-5 Networking diagram (IPv6)

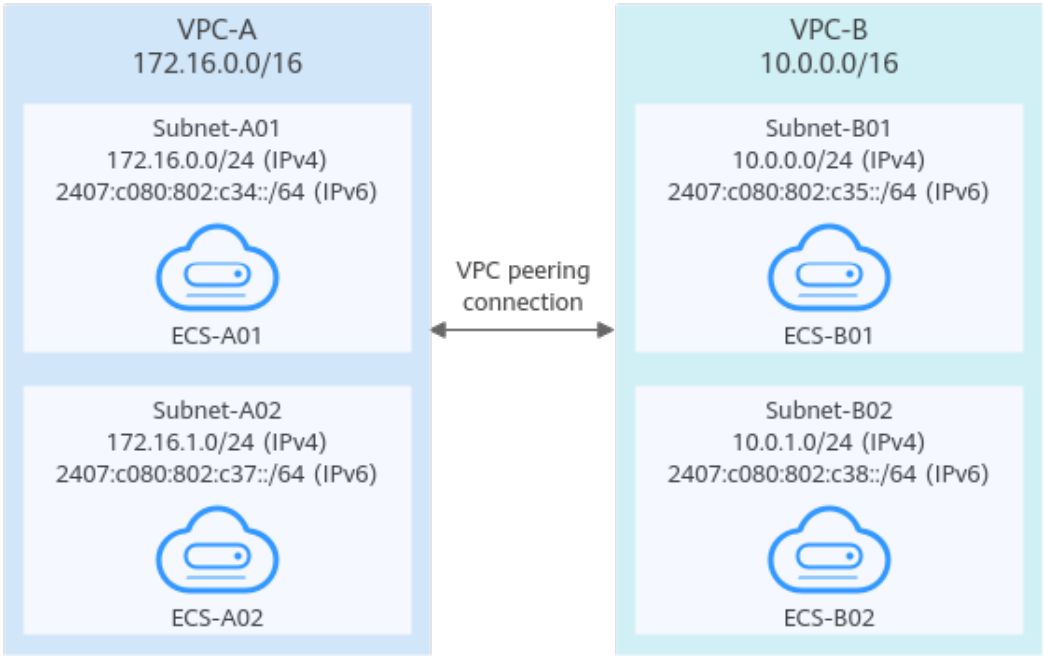


Table 8-6 Resource planning details (IPv6)

VPC Name	VPC CIDR Block	Subnet Name	Subnet CIDR Block	Subnet Route Table	ECS Name	Security Group	Private IP Address
VPC-A	172.16.0.0/16	Subnet-A01	• IPv4: 172.16.0.0/24 • IPv6: 2407:c080:802:c34::/64	rtb-VPC-A	ECS-A01	sg-web: general-purpose web server	• IPv4: 172.16.0.111 • IPv6: 2407:c080:802:c34:a925:f12e:cfa0:8edb
		Subnet-A02	• IPv4: 172.16.1.0/24 • IPv6: 2407:c080:802:c37::/64	rtb-VPC-A	ECS-A02		• IPv4: 172.16.1.91 • IPv6: 2407:c080:802:c37:594b:4c0f:2fcd:8b72

VPC Name	VPC CIDR Block	Subnet Name	Subnet CIDR Block	Subnet Route Table	ECS Name	Security Group	Private IP Address
VPC-B	10.0.0.0/16	Subnet-B01	• IPv4: 10.0.0.0/24 • IPv6: 2407:c080:802:c35::/64	rtb-VPC-B	ECS-B01		• IPv4: 10.0.0.139 • IPv6: 2407:c080:802:c35:493:33f4:4531:5162
		Subnet-B02	• IPv4: 10.0.1.0/24 • IPv6: 2407:c080:802:c38::/64	rtb-VPC-B	ECS-B02		• IPv4: 10.0.1.167 • IPv6: 2407:c080:802:c38:b9a9:aa03:2700:c1cf

Table 8-7 Peering relationships (IPv6)

Peering Relationship	Peering Connection Name	Local VPC	Peer VPC
VPC-A is peered with VPC-B.	Peering-AB	VPC-A	VPC-B

After the VPC peering connection is created, add the following routes to the route tables of the local and peer VPCs:

Table 8-8 VPC route tables (IPv6)

Route Table	Destination	Next Hop	Route Type	Description
rtb-VPC-A	172.16.0.0/24	Local	System	Local routes are automatically added for communications within a VPC.

Route Table	Destination	Next Hop	Route Type	Description
	2407:c080:802:c34::/64	Local	System	
	172.16.1.0/24	Local	System	
	2407:c080:802:c37::/64	Local	System	
	10.0.0.0/16 (VPC-B)	Peering-AB	Custom	Add a route with the CIDR block of VPC-B as the destination and Peering-AB as the next hop for IPv4 communication.
	2407:c080:802:c35::/64 (Subnet-B01)	Peering-AB	Custom	Add routes with the IPv6 CIDR blocks of Subnet-B01 and Subnet-B02 as the destinations and Peering-AB as the next hop for IPv6 communication.
	2407:c080:802:c38::/64 (Subnet-B02)	Peering-AB	Custom	
rtb-VPC-B	10.0.0.0/24	Local	System	Local routes are automatically added for communications within a VPC.
	2407:c080:802:c35::/64	Local	System	
	10.0.1.0/24	Local	System	
	2407:c080:802:c38::/64	Local	System	
	172.16.0.0/16 (VPC-A)	Peering-AB	Custom	Add a route with the CIDR block of VPC-A as the destination and Peering-AB as the next hop for IPv4 communication.
	2407:c080:802:c34::/64 (Subnet-A01)	Peering-AB	Custom	Add routes with the IPv6 CIDR blocks of Subnet-A01 and Subnet-A02 as the destinations and Peering-AB as the next hop for IPv6 communication.
	2407:c080:802:c37::/64 (Subnet-A02)	Peering-AB	Custom	

NOTE

You can view IPv6 addresses of VPC subnets on the management console. To enable communication between entire CIDR blocks of two VPCs, you need to add routes with IPv6 CIDR blocks of all subnets in the VPCs as the destinations one by one.

Multiple VPCs Peered Together (IPv4)

If multiple VPCs need to communicate with each other, their CIDR blocks cannot overlap and you need to create a VPC peering connection between every two VPCs.

- For details about resource planning, see **Table 8-9**.
- For details about VPC peering relationships, see **Table 8-10**.

Figure 8-6 Networking diagram (IPv4)

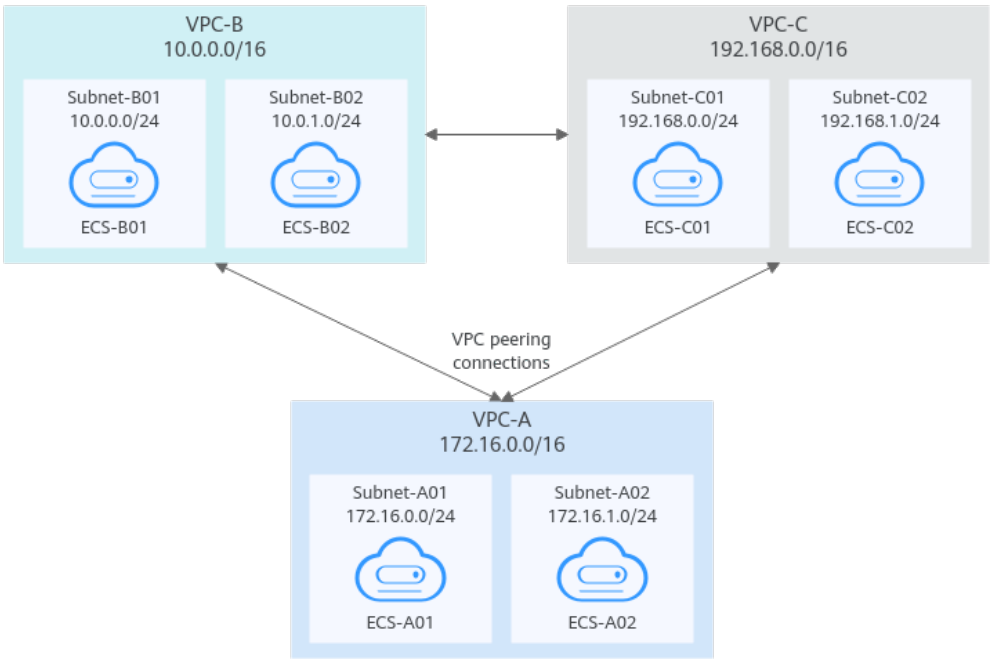


Table 8-9 Resource planning details (IPv4)

VPC Name	VPC CIDR Block	Subnet Name	Subnet CIDR Block	Subnet Route Table	ECS Name	Security Group	Private IP Address
VPC-A	172.16.0.0/16	Subnet-A01	172.16.0.0/24	rtb-VPC-A	ECS-A01	sg-web: general-purpose web server	172.16.0.111
		Subnet-A02	172.16.1.0/24	rtb-VPC-A	ECS-A02		172.16.1.91
VPC-B	10.0.0.0/16	Subnet-B01	10.0.0.0/24	rtb-VPC-B	ECS-B01		10.0.0.139

VPC Name	VPC CIDR Block	Subnet Name	Subnet CIDR Block	Subnet Route Table	ECS Name	Security Group	Private IP Address
		Subnet-B02	10.0.1.0/24	rtb-VPC-B	ECS-B02		10.0.1.167
VPC-C	192.168.0.0/16	Subnet-C01	192.168.0.0/24	rtb-VPC-C	ECS-C01		192.168.0.194
		Subnet-C02	192.168.1.0/24	rtb-VPC-C	ECS-C02		192.168.1.200

Table 8-10 Peering relationships (IPv4)

Peering Relationship	Peering Connection Name	Local VPC	Peer VPC
VPC-A is peered with VPC-B.	Peering-AB	VPC-A	VPC-B
VPC-A is peered with VPC-C.	Peering-AC	VPC-A	VPC-C
VPC-B is peered with VPC-C.	Peering-BC	VPC-B	VPC-C

After the VPC peering connections are created, add the following routes to the route tables of the local and peer VPCs:

Table 8-11 VPC route tables (IPv4)

Route Table	Destination	Next Hop	Route Type	Description
rtb-VPC-A	172.16.0.0/24	Local	System	Local routes are automatically added for communications within a VPC.
	172.16.1.0/24	Local	System	
	10.0.0.0/16 (VPC-B)	Peering-AB	Custom	Add a route with the CIDR block of VPC-B as the destination and Peering-AB as the next hop.

Route Table	Destination	Next Hop	Route Type	Description
	192.168.0.0/16 (VPC-C)	Peering-AC	Custom	Add a route with the CIDR block of VPC-C as the destination and Peering-AC as the next hop.
rtb-VPC-B	10.0.0.0/24	Local	System	Local routes are automatically added for communications within a VPC.
	10.0.1.0/24	Local	System	
	172.16.0.0/16 (VPC-A)	Peering-AB	Custom	Add a route with the CIDR block of VPC-A as the destination and Peering-AB as the next hop.
	192.168.0.0/16 (VPC-C)	Peering-BC	Custom	Add a route with the CIDR block of VPC-C as the destination and Peering-BC as the next hop.
rtb-VPC-C	192.168.0.0/24	Local	System	Local routes are automatically added for communications within a VPC.
	192.168.1.0/24	Local	System	
	172.16.0.0/16 (VPC-A)	Peering-AC	Custom	Add a route with the CIDR block of VPC-A as the destination and Peering-AC as the next hop.
	10.0.0.0/16 (VPC-B)	Peering-BC	Custom	Add a route with the CIDR block of VPC-B as the destination and Peering-BC as the next hop.

NOTE

If the destination of a route in a route table of a VPC is set to the CIDR block of a peer VPC and the CIDR blocks of the two VPCs do not overlap, the VPCs can have full access to each other's resources.

Multiple VPCs Peered Together Through Transitive Peering Connections (IPv4)

VPC peering connections are transitive. As shown in **Figure 8-7**, there is a VPC peering connection between VPC-A and VPC-B, and between VPC-A and VPC-C. To enable communication between VPC-B and VP-C, you can use either of the following methods:

- Create a VPC peering connection between VPC-B and VPC-C. For details, see **Multiple VPCs Peered Together (IPv4)**.

- Add routes to direct traffic between VPC-B and VPC-C based on VPC-A. For details, see **Table 8-14**.

Figure 8-7 Transitive VPC peering connections

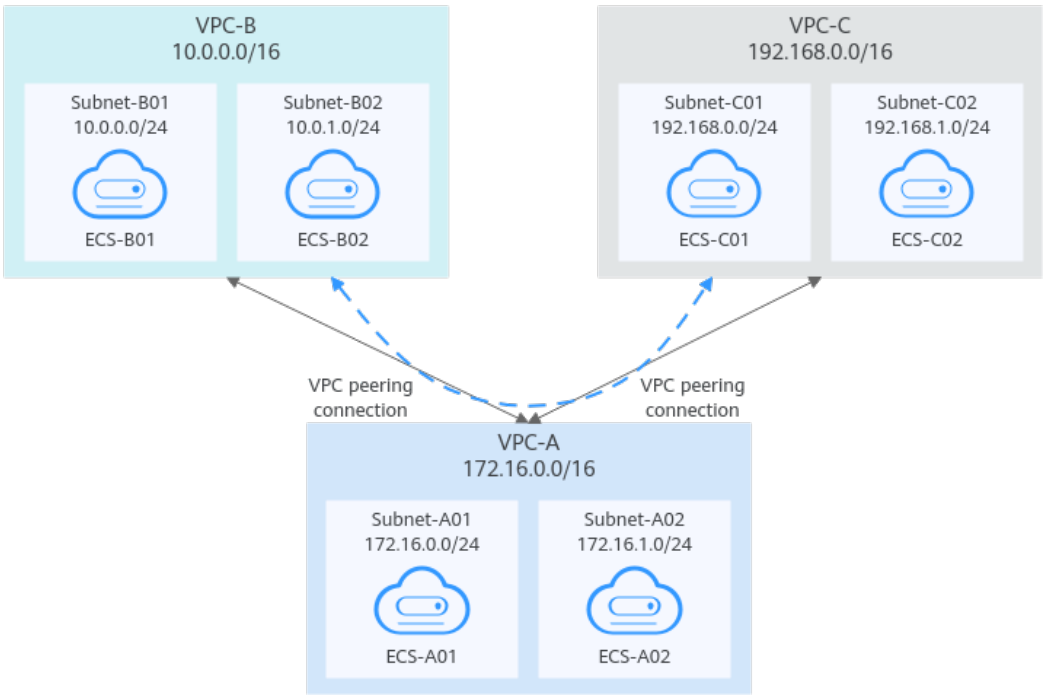


Table 8-12 Resource planning details (IPv4)

VPC Name	VPC CIDR Block	Subnet Name	Subnet CIDR Block	Subnet Route Table	ECS Name	Security Group	Private IP Address
VPC-A	172.16.0.0/16	Subnet-A01	172.16.0.0/24	rtb-VPC-A	ECS-A01	sg-web: general-purpose web server	172.16.0.111
		Subnet-A02	172.16.1.0/24	rtb-VPC-A	ECS-A02		172.16.1.91
VPC-B	10.0.0.0/16	Subnet-B01	10.0.0.0/24	rtb-VPC-B	ECS-B01		10.0.0.139
		Subnet-B02	10.0.1.0/24	rtb-VPC-B	ECS-B02		10.0.1.167
VPC-C	192.168.0.0/16	Subnet-C01	192.168.0.0/24	rtb-VPC-C	ECS-C01		192.168.0.194
		Subnet-C02	192.168.1.0/24	rtb-VPC-C	ECS-C02		192.168.1.200

Table 8-13 Peering relationships (IPv4)

Peering Relationship	Peering Connection Name	Local VPC	Peer VPC
VPC-A is peered with VPC-B.	Peering-AB	VPC-A	VPC-B
VPC-A is peered with VPC-C.	Peering-AC	VPC-A	VPC-C

After the VPC peering connections are created, add the following routes to the route tables of the local and peer VPCs:

Table 8-14 VPC route tables (IPv4)

Route Table	Destination	Next Hop	Route Type	Description
rtb-VPC-A	172.16.0.0/24	Local	System	Local routes are automatically added for communications within a VPC.
	172.16.1.0/24	Local	System	
	10.0.0.0/16 (VPC-B)	Peering-AB	Custom	Add a route with the CIDR block of VPC-B as the destination and Peering-AB as the next hop.
	192.168.0.0/16 (VPC-C)	Peering-AC	Custom	Add a route with the CIDR block of VPC-C as the destination and Peering-AC as the next hop.
rtb-VPC-B	10.0.0.0/24	Local	System	Local routes are automatically added for communications within a VPC.
	10.0.1.0/24	Local	System	
	172.16.0.0/16 (VPC-A)	Peering-AB	Custom	Add a route with the CIDR block of VPC-A as the destination and Peering-AB as the next hop.
	192.168.0.0/16 (VPC-C)	Peering-AB	Custom	Add a route with the CIDR block of VPC-C as the destination and Peering-AB as the next hop.
rtb-VPC-C	192.168.0.0/24	Local	System	Local routes are automatically added for communications within a VPC.

Route Table	Destination	Next Hop	Route Type	Description
	192.168.1.0/24	Local	System	
	172.16.0.0/16 (VPC-A)	Peering-AC	Custom	Add a route with the CIDR block of VPC-A as the destination and Peering-AC as the next hop.
	10.0.0.0/16 (VPC-B)	Peering-AC	Custom	Add a route with the CIDR block of VPC-B as the destination and Peering-AC as the next hop.

NOTE

If the destination of a route in a route table of a VPC is set to the CIDR block of a peer VPC and the CIDR blocks of the two VPCs do not overlap, the VPCs can have full access to each other's resources.

Multiple VPCs Peered Together (IPv6)

If multiple VPCs need to communicate with each other, you need to create a VPC peering connection between every two VPCs. In this example, subnets in VPC-A, VPC-B, and VPC-C have IPv6 CIDR blocks and the IPv4 CIDR blocks of VPC-A, VPC-B, and VPC-C cannot overlap.

- For details about resource planning, see **Table 8-15**.
- For details about VPC peering relationships, see **Table 8-16**.

Figure 8-8 Networking diagram (IPv6)

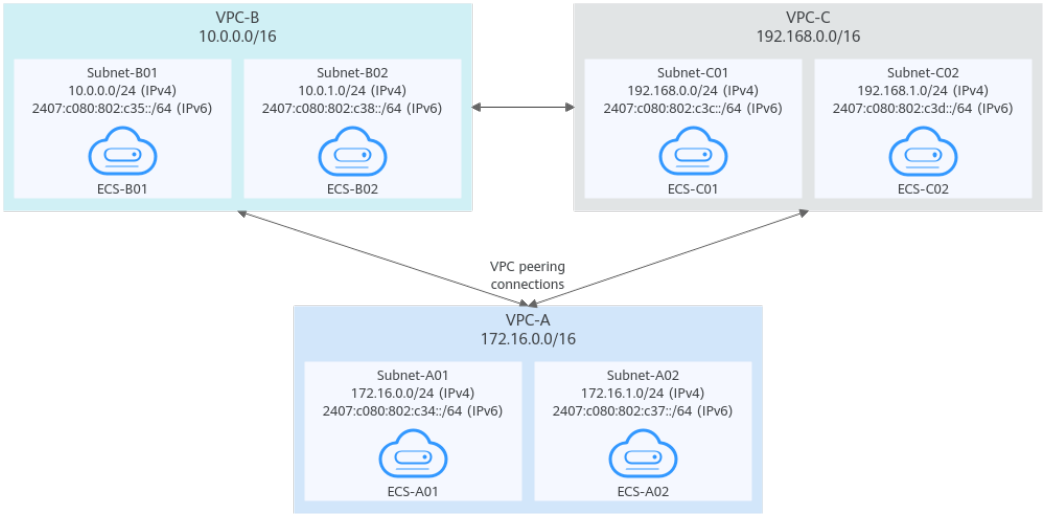


Table 8-15 Resource planning details (IPv6)

VPC Name	VPC CIDR Block	Subnet Name	Subnet CIDR Block	Subnet Route Table	ECS Name	Security Group	Private IP Address
VPC-A	172.16.0.0/16	Subnet-A01	• IPv4: 172.16.0.0/24 • IPv6: 2407:c080:802:c34::/64	rtb-VPC-A	ECS-A01	sg-web: general-purpose web server	• IPv4: 172.16.0.111 • IPv6: 2407:c080:802:c34:a925:f12e:cfa0:8edb
		Subnet-A02	• IPv4: 172.16.1.0/24 • IPv6: 2407:c080:802:c37::/64	rtb-VPC-A	ECS-A02		• IPv4: 172.16.1.91 • IPv6: 2407:c080:802:c37:594b:4c0f:2fcd:8b72
VPC-B	10.0.0.0/16	Subnet-B01	• IPv4: 10.0.0.0/24 • IPv6: 2407:c080:802:c35::/64	rtb-VPC-B	ECS-B01		• IPv4: 10.0.0.139 • IPv6: 2407:c080:802:c35:493:33f4:4531:5162
		Subnet-B02	• IPv4: 10.0.1.0/24 • IPv6: 2407:c080:802:c38::/64	rtb-VPC-B	ECS-B02		• IPv4: 10.0.1.167 • IPv6: 2407:c080:802:c38:b9a9:aa03:2700:c1cf

VPC Name	VPC CIDR Block	Subnet Name	Subnet CIDR Block	Subnet Route Table	ECS Name	Security Group	Private IP Address
VPC-C	192.168.0.0/16	Subnet-C01	• IPv4: 192.168.0.0/24 • IPv6: 2407:c080:802:c3c::/64	rtb-VPC-C	ECS-C01		• IPv4: 192.168.0.194 • IPv6: 2407:c080:802:c3c:d2f3:d891:24f5:f4af
		Subnet-C02	• IPv4: 192.168.1.0/24 • IPv6: 2407:c080:802:c3d::/64	rtb-VPC-C	ECS-C02		• IPv4: 192.168.1.200 • IPv6: 2407:c080:802:c3d:e9ca:169a:390c:74d1

Table 8-16 Peering relationships (IPv6)

Peering Relationship	Peering Connection Name	Local VPC	Peer VPC
VPC-A is peered with VPC-B.	Peering-AB	VPC-A	VPC-B
VPC-A is peered with VPC-C.	Peering-AC	VPC-A	VPC-C
VPC-B is peered with VPC-C.	Peering-BC	VPC-B	VPC-C

After the VPC peering connections are created, add the following routes to the route tables of the local and peer VPCs:

Table 8-17 VPC route tables (IPv6)

Route Table	Destination	Next Hop	Route Type	Description
rtb-VPC-A	172.16.0.0/24	Local	System	Local routes are automatically added for communications within a VPC.
	2407:c080:802:c34::/64	Local	System	
	172.16.1.0/24	Local	System	
	2407:c080:802:c37::/64	Local	System	
	10.0.0.0/16 (VPC-B)	Peering-AB	Custom	Add a route with the CIDR block of VPC-B as the destination and Peering-AB as the next hop for IPv4 communication.
	2407:c080:802:c35::/64 (Subnet-B01)	Peering-AB	Custom	Add routes with the IPv6 CIDR blocks of Subnet-B01 and Subnet-B02 as the destinations and Peering-AB as the next hop for IPv6 communication.
	2407:c080:802:c38::/64 (Subnet-B02)	Peering-AB	Custom	
	192.168.0.0/16 (VPC-C)	Peering-AC	Custom	Add a route with the CIDR block of VPC-C as the destination and Peering-AC as the next hop for IPv4 communication.
	2407:c080:802:c3c::/64 (Subnet-C01)	Peering-AC	Custom	Add routes with the IPv6 CIDR blocks of Subnet-C01 and Subnet-C02 as the destinations and Peering-AC as the next hop for IPv6 communication.
	2407:c080:802:c3d::/64 (Subnet-C02)	Peering-AC	Custom	
rtb-VPC-B	10.0.0.0/24	Local	System	Local routes are automatically added for communications within a VPC.
	2407:c080:802:c35::/64	Local	System	
	10.0.1.0/24	Local	System	
	2407:c080:802:c38::/64	Local	System	

Route Table	Destination	Next Hop	Route Type	Description
	172.16.0.0/16 (VPC-A)	Peering-AB	Custom	Add a route with the CIDR block of VPC-A as the destination and Peering-AB as the next hop for IPv4 communication.
	2407:c080:802:c34::/64 (Subnet-A01)	Peering-AB	Custom	Add routes with the IPv6 CIDR blocks of Subnet-A01 and Subnet-A02 as the destinations and Peering-AB as the next hop for IPv6 communication.
	2407:c080:802:c37::/64 (Subnet-A02)	Peering-AB	Custom	
	192.168.0.0/16 (VPC-C)	Peering-BC	Custom	Add a route with the CIDR block of VPC-C as the destination and Peering-BC as the next hop for IPv4 communication.
	2407:c080:802:c3c::/64 (Subnet-C01)	Peering-BC	Custom	Add routes with the IPv6 CIDR blocks of Subnet-C01 and Subnet-C02 as the destinations and Peering-BC as the next hop for IPv6 communication.
	2407:c080:802:c3d::/64 (Subnet-C02)	Peering-BC	Custom	
rtb-VPC-C	192.168.0.0/24	Local	System	Local routes are automatically added for communications within a VPC.
	2407:c080:802:c3c::/64	Local	System	
	192.168.1.0/24	Local	System	
	2407:c080:802:c3d::/64	Local	System	
	172.16.0.0/16 (VPC-A)	Peering-AC	Custom	Add a route with the CIDR block of VPC-A as the destination and Peering-AC as the next hop for IPv4 communication.
	2407:c080:802:c34::/64 (Subnet-A01)	Peering-AC	Custom	Add routes with the IPv6 CIDR blocks of Subnet-A01 and Subnet-A02 as the destinations and Peering-AC as the next hop for IPv6 communication.
	2407:c080:802:c37::/64 (Subnet-A02)	Peering-AC	Custom	

Route Table	Destination	Next Hop	Route Type	Description
	10.0.0.0/16 (VPC-B)	Peering-BC	Custom	Add a route with the CIDR block of VPC-B as the destination and Peering-BC as the next hop for IPv4 communication.
	2407:c080:802:c35::/64 (Subnet-B01)	Peering-BC	Custom	Add routes with the IPv6 CIDR blocks of Subnet-B01 and Subnet-B02 as the destinations and Peering-BC as the next hop for IPv6 communication.
	2407:c080:802:c38::/64 (Subnet-B02)	Peering-BC	Custom	

NOTE

You can view IPv6 addresses of VPC subnets on the management console. To enable communication between entire CIDR blocks of two VPCs, you need to add routes with IPv6 CIDR blocks of all subnets in the VPCs as the destinations one by one.

One Central VPC Peered with Two VPCs (IPv4)

Create Peering-AB between VPC-A and VPC-B, and Peering-AC between VPC-A and VPC-C. The three VPCs do not have overlapping CIDR blocks.

- For details about resource planning, see **Table 8-18**.
- For details about VPC peering relationships, see **Table 8-19**.

Figure 8-9 Networking diagram (IPv4)

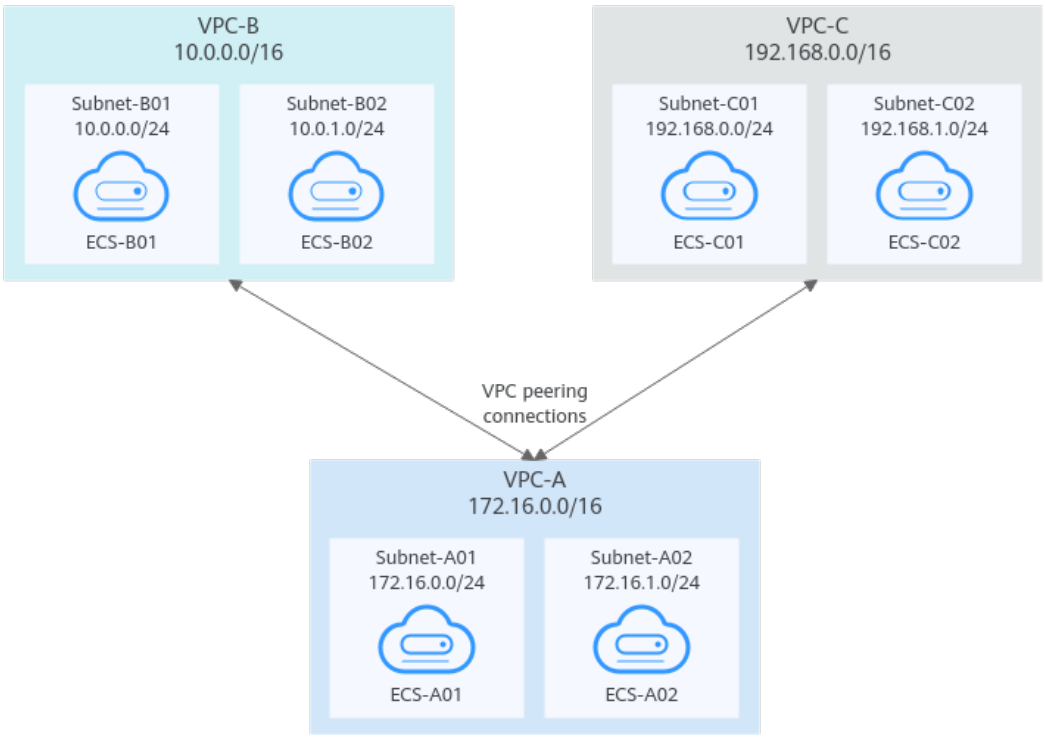


Table 8-18 Resource planning details (IPv4)

VPC Name	VPC CIDR Block	Subnet Name	Subnet CIDR Block	Subnet Route Table	ECS Name	Security Group	Private IP Address
VPC-A	172.16.0.0/16	Subnet-A01	172.16.0.0/24	rtb-VPC-A	ECS-A01	sg-web: general-purpose web server	172.16.0.111
		Subnet-A02	172.16.1.0/24	rtb-VPC-A	ECS-A02		172.16.1.91
VPC-B	10.0.0.0/16	Subnet-B01	10.0.0.0/24	rtb-VPC-B	ECS-B01		10.0.0.139
		Subnet-B02	10.0.1.0/24	rtb-VPC-B	ECS-B02		10.0.1.167
VPC-C	192.168.0.0/16	Subnet-C01	192.168.0.0/24	rtb-VPC-C	ECS-C01		192.168.0.194
		Subnet-C02	192.168.1.0/24	rtb-VPC-C	ECS-C02		192.168.1.200

Table 8-19 Peering relationships (IPv4)

Peering Relationship	Peering Connection Name	Local VPC	Peer VPC
VPC-A is peered with VPC-B.	Peering-AB	VPC-A	VPC-B
VPC-A is peered with VPC-C.	Peering-AC	VPC-A	VPC-C

After the VPC peering connections are created, add the following routes to the route tables of the local and peer VPCs:

Table 8-20 VPC route table details (IPv4)

Route Table	Destination	Next Hop	Route Type	Description
rtb-VPC-A	172.16.0.0/24	Local	System	Local routes are automatically added for communications within a VPC.
	172.16.1.0/24	Local	System	
	10.0.0.0/16 (VPC-B)	Peering-AB	Custom	Add a route with the CIDR block of VPC-B as the destination and Peering-AB as the next hop.
	192.168.0.0/16 (VPC-C)	Peering-AC	Custom	Add a route with the CIDR block of VPC-C as the destination and Peering-AC as the next hop.
rtb-VPC-B	10.0.0.0/24	Local	System	Local routes are automatically added for communications within a VPC.
	10.0.1.0/24	Local	System	
	172.16.0.0/16 (VPC-A)	Peering-AB	Custom	Add a route with the CIDR block of VPC-A as the destination and Peering-AB as the next hop.
rtb-VPC-C	192.168.0.0/24	Local	System	Local routes are automatically added for communications within a VPC.
	192.168.1.0/24	Local	System	

Route Table	Destination	Next Hop	Route Type	Description
	172.16.0.0/16 (VPC-A)	Peering-AC	Custom	Add a route with the CIDR block of VPC-A as the destination and Peering-AC as the next hop.

NOTE

If the destination of a route in a route table of a VPC is set to the CIDR block of a peer VPC and the CIDR blocks of the two VPCs do not overlap, the VPCs can have full access to each other's resources.

One Central VPC Peered with Two VPCs (IPv6)

Create Peering-AB between VPC-A and VPC-B, and Peering-AC between VPC-A and VPC-C. Each VPC has IPv6 subnets. The IPv4 CIDR blocks of the three VPCs do not overlap with each other.

- For details about resource planning, see **Table 8-21**.
- For details about VPC peering relationships, see **Table 8-22**.

Figure 8-10 Networking diagram (IPv6)

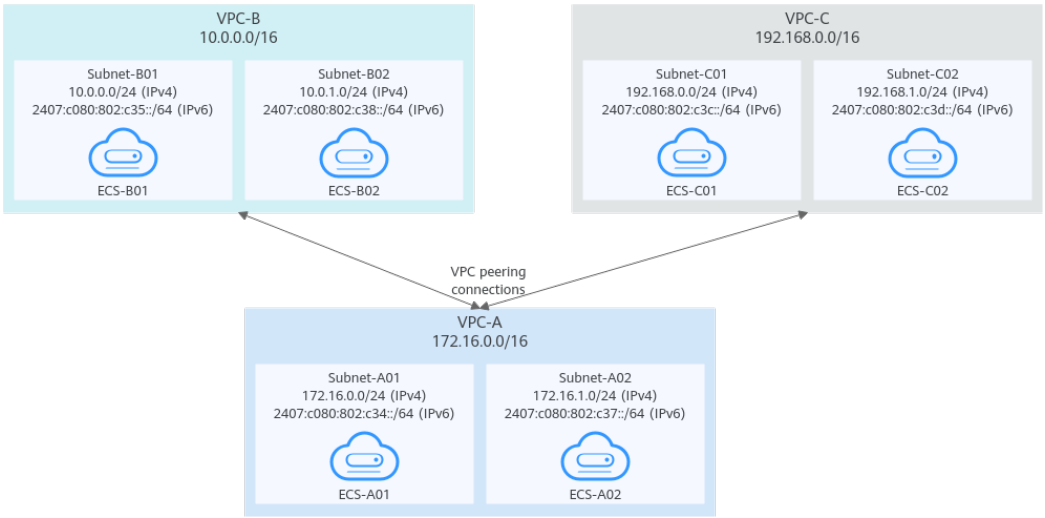


Table 8-21 Resource planning details (IPv6)

VPC Name	VPC CIDR Block	Subnet Name	Subnet CIDR Block	Subnet Route Table	ECS Name	Security Group	Private IP Address
VPC-A	172.16.0.0/16	Subnet-A01	• IPv4: 172.16.0.0/24 • IPv6: 2407:c080:802:c34::/64	rtb-VPC-A	ECS-A01	sg-web: general-purpose web server	• IPv4: 172.16.0.111 • IPv6: 2407:c080:802:c34:a925:f12e:cfa0:8edb
		Subnet-A02	• IPv4: 172.16.1.0/24 • IPv6: 2407:c080:802:c37::/64	rtb-VPC-A	ECS-A02		• IPv4: 172.16.1.91 • IPv6: 2407:c080:802:c37:594b:4c0f:2fcd:8b72
VPC-B	10.0.0.0/16	Subnet-B01	• IPv4: 10.0.0.0/24 • IPv6: 2407:c080:802:c35::/64	rtb-VPC-B	ECS-B01		• IPv4: 10.0.0.139 • IPv6: 2407:c080:802:c35:493:33f4:4531:5162
		Subnet-B02	• IPv4: 10.0.1.0/24 • IPv6: 2407:c080:802:c38::/64	rtb-VPC-B	ECS-B02		• IPv4: 10.0.1.167 • IPv6: 2407:c080:802:c38:b9a9:aa03:2700:c1cf

VPC Name	VPC CIDR Block	Subnet Name	Subnet CIDR Block	Subnet Route Table	ECS Name	Security Group	Private IP Address
VPC-C	192.168.0.0/16	Subnet-C01	• IPv4: 192.168.0.0/24 • IPv6: 2407:c080:802:c3c::/64	rtb-VPC-C	ECS-C01		• IPv4: 192.168.0.194 • IPv6: 2407:c080:802:c3c:d2f3:d891:24f5:f4af
		Subnet-C02	• IPv4: 192.168.1.0/24 • IPv6: 2407:c080:802:c3d::/64	rtb-VPC-C	ECS-C02		• IPv4: 192.168.1.200 • IPv6: 2407:c080:802:c3d:e9ca:169a:390c:74d1

Table 8-22 Peering relationships (IPv6)

Peering Relationship	Peering Connection Name	Local VPC	Peer VPC
VPC-A is peered with VPC-B.	Peering-AB	VPC-A	VPC-B
VPC-A is peered with VPC-C.	Peering-AC	VPC-A	VPC-C

After the VPC peering connections are created, add the following routes to the route tables of the local and peer VPCs:

Table 8-23 VPC route table details (IPv6)

Route Table	Destination	Next Hop	Route Type	Description
rtb-VPC-A	172.16.0.0/24	Local	System	Local routes are automatically added for communications within a VPC.
	2407:c080:802:c34::/64	Local	System	
	172.16.1.0/24	Local	System	
	2407:c080:802:c37::/64	Local	System	
	10.0.0.0/16 (VPC-B)	Peering-AB	Custom	Add a route with the CIDR block of VPC-B as the destination and Peering-AB as the next hop for IPv4 communication.
	2407:c080:802:c35::/64 (Subnet-B01)	Peering-AB	Custom	Add routes with the IPv6 CIDR blocks of Subnet-B01 and Subnet-B02 as the destinations and Peering-AB as the next hop for IPv6 communication.
	2407:c080:802:c38::/64 (Subnet-B02)	Peering-AB	Custom	
	192.168.0.0/16 (VPC-C)	Peering-AC	Custom	Add a route with the CIDR block of VPC-C as the destination and Peering-AC as the next hop for IPv4 communication.
	2407:c080:802:c3c::/64 (Subnet-C01)	Peering-AC	Custom	Add routes with the IPv6 CIDR blocks of Subnet-C01 and Subnet-C02 as the destinations and Peering-AC as the next hop for IPv6 communication.
	2407:c080:802:c3d::/64 (Subnet-C02)	Peering-AC	Custom	
rtb-VPC-B	10.0.0.0/24	Local	System	Local routes are automatically added for communications within a VPC.
	2407:c080:802:c35::/64	Local	System	
	10.0.1.0/24	Local	System	
	2407:c080:802:c38::/64	Local	System	

Route Table	Destination	Next Hop	Route Type	Description
	172.16.0.0/16 (VPC-A)	Peering-AB	Custom	Add a route with the CIDR block of VPC-A as the destination and Peering-AB as the next hop for IPv4 communication.
	2407:c080:802:c34::/64 (Subnet-A01)	Peering-AB	Custom	Add routes with the IPv6 CIDR blocks of Subnet-A01 and Subnet-A02 as the destinations and Peering-AB as the next hop for IPv6 communication.
	2407:c080:802:c37::/64 (Subnet-A02)	Peering-AB	Custom	
rtb-VPC-C	192.168.0.0/24	Local	System	Local routes are automatically added for communications within a VPC.
	2407:c080:802:c3c::/64	Local	System	
	192.168.1.0/24	Local	System	
	2407:c080:802:c3d::/64	Local	System	
	172.16.0.0/16 (VPC-A)	Peering-AC	Custom	Add a route with the CIDR block of VPC-A as the destination and Peering-AC as the next hop for IPv4 communication.
	2407:c080:802:c34::/64 (Subnet-A01)	Peering-AC	Custom	Add routes with the IPv6 CIDR blocks of Subnet-A01 and Subnet-A02 as the destinations and Peering-AC as the next hop for IPv6 communication.
	2407:c080:802:c37::/64 (Subnet-A02)	Peering-AC	Custom	

NOTE

You can view IPv6 addresses of VPC subnets on the management console. To enable communication between entire CIDR blocks of two VPCs, you need to add routes with IPv6 CIDR blocks of all subnets in the VPCs as the destinations one by one.

One Central VPC with Primary and Secondary CIDR Blocks Peered with Two VPCs (IPv4)

Create Peering-AB between VPC-A and VPC-B, and Peering-AC between VPC-A and VPC-C. VPC-A has both primary and secondary CIDR blocks. The three VPCs do not have overlapping CIDR blocks.

- For details about resource planning, see **Table 8-24**.
- For details about VPC peering relationships, see **Table 8-25**.

Figure 8-11 Networking diagram (IPv4)

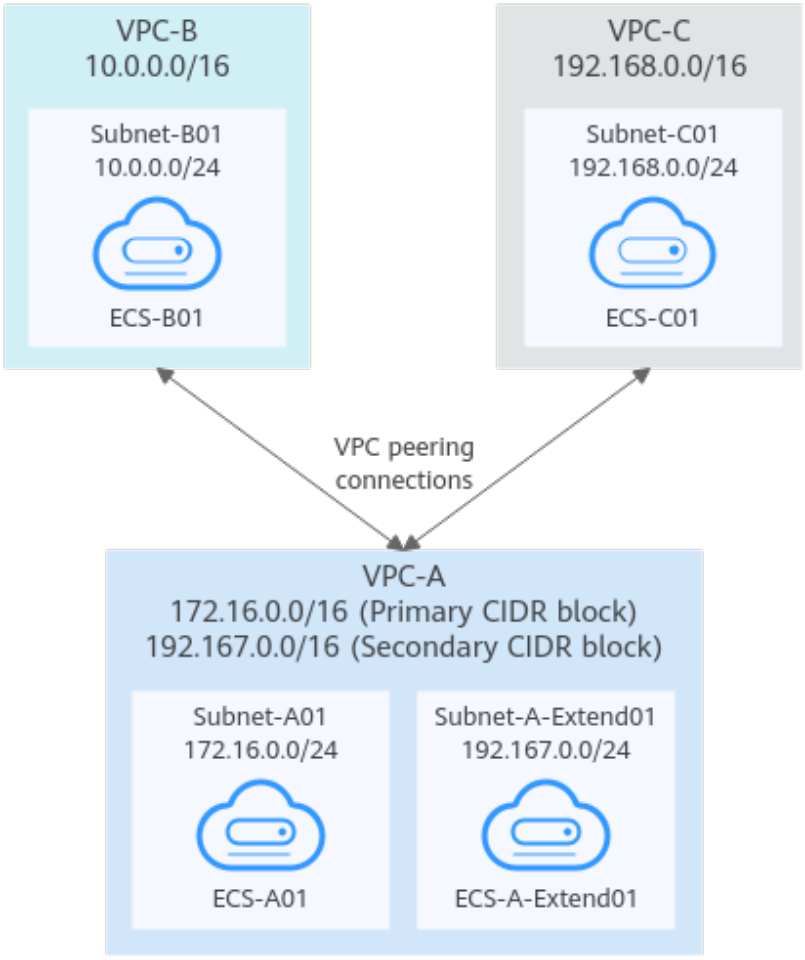


Table 8-24 Resource planning details

VPC Name	VPC CIDR Block	Subnet Name	Subnet CIDR Block	Subnet Route Table	ECS Name	Security Group	Private IP Address
VPC-A	Primary CIDR block: 172.16.0.0/16	Subnet-A01	172.16.0.0/24	rtb-VPC-A	ECS-A01	sg-web: general-purpose web server	172.16.0.111
	Secondary CIDR block: 192.167.0.0/16	Subnet-A-Extend01	192.167.0.0/24	rtb-VPC-A	ECS-A-Extend01		192.167.0.100
VPC-B	10.0.0.0/16	Subnet-B01	10.0.0.0/24	rtb-VPC-B	ECS-B01		10.0.0.139
VPC-C	192.168.0.0/16	Subnet-C01	192.168.0.0/24	rtb-VPC-C	ECS-C01		192.168.0.194

Table 8-25 Peering relationships (IPv4)

Peering Relationship	Peering Connection Name	Local VPC	Peer VPC
VPC-A is peered with VPC-B.	Peering-AB	VPC-A	VPC-B
VPC-A is peered with VPC-C.	Peering-AC	VPC-A	VPC-C

After the VPC peering connections are created, add the following routes to the route tables of the local and peer VPCs:

Table 8-26 VPC route table details (IPv4)

Route Table	Destination	Next Hop	Route Type	Description
rtb-VPC-A	172.16.0.0/24	Local	System	Local routes are automatically added for communications within a VPC.
	192.167.0.0/24	Local	System	
	10.0.0.0/16 (VPC-B)	Peering-AB	Custom	Add a route with the CIDR block of VPC-B as the destination and Peering-AB as the next hop.
	192.168.0.0/16 (VPC-C)	Peering-AC	Custom	Add a route with the CIDR block of VPC-C as the destination and Peering-AC as the next hop.
rtb-VPC-B	10.0.0.0/24	Local	System	Local routes are automatically added for communications within a VPC.
	172.16.0.0/16 (Primary CIDR block of VPC-A)	Peering-AB	Custom	Add routes with the primary and secondary CIDR blocks of VPC-A as the destinations and Peering-AB as the next hop.
	192.167.0.0/16 (Secondary CIDR block of VPC-A)	Peering-AB	Custom	
rtb-VPC-C	192.168.0.0/24	Local	System	Local routes are automatically added for communications within a VPC.
	172.16.0.0/16 (Primary CIDR block of VPC-A)	Peering-AC	Custom	Add routes with the primary and secondary CIDR blocks of VPC-A as the destinations and Peering-AC as the next hop.
	192.167.0.0/16 (Secondary CIDR block of VPC-A)	Peering-AC	Custom	

NOTE

If the destination of a route in a route table of a VPC is set to the CIDR block of a peer VPC and the CIDR blocks of the two VPCs do not overlap, the VPCs can have full access to each other's resources.

One Central VPC Peered with Multiple VPCs (IPv4)

Create a VPC peering connection between VPC-A and VPC-B, between VPC-A and VPC-C, between VPC-A and VPC-D, between VPC-A and VPC-E, between VPC-A

and VPC-F, and between VPC-A and VPC-G. The CIDR blocks of these VPCs do not overlap.

- For details about resource planning, see **Table 8-27**.
- For details about VPC peering relationships, see **Table 8-28**.

Figure 8-12 Networking diagram (IPv4)

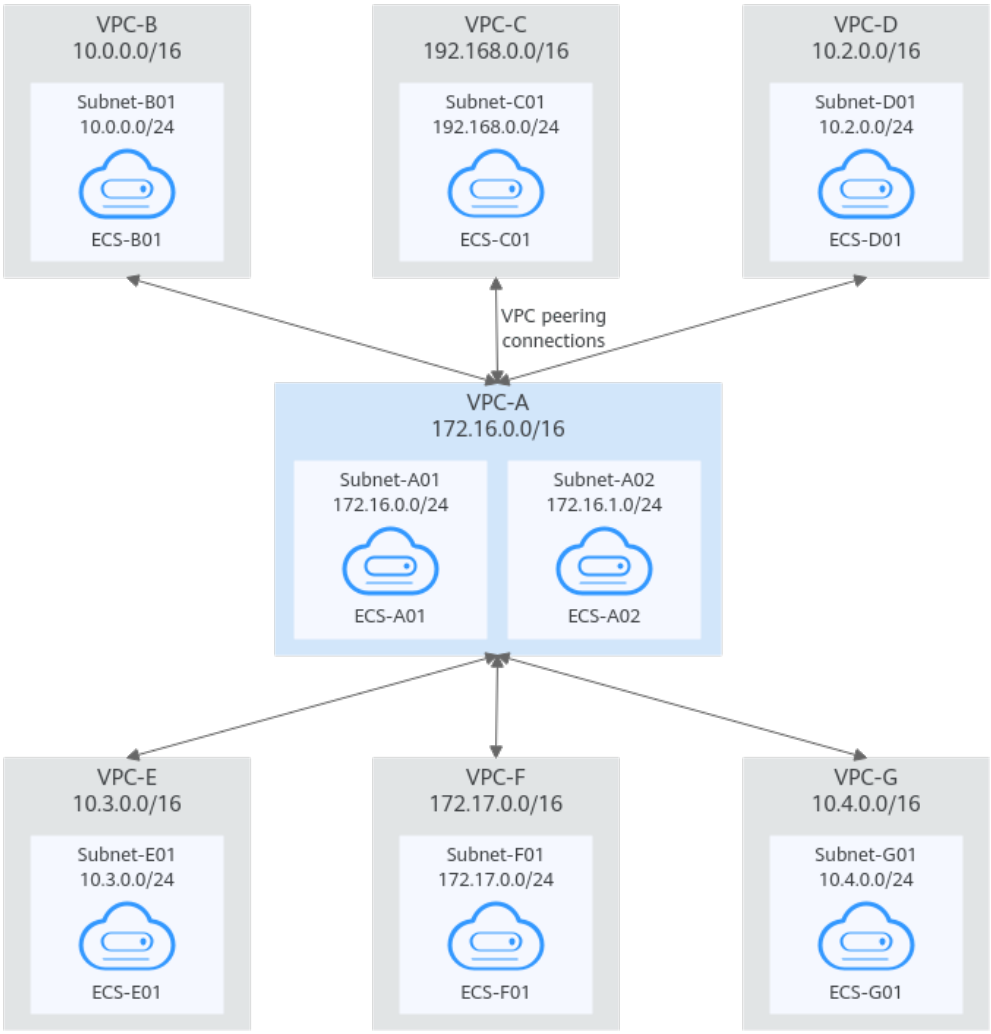


Table 8-27 Resource planning details (IPv4)

VPC Name	VPC CIDR Block	Subnet Name	Subnet CIDR Block	Subnet Route Table	ECS Name	Security Group	Private IP Address
VPC-A	172.16.0.0/16	Subnet-A01	172.16.0.0/24	rtb-VPC-A	ECS-A01	sg-web: general-purpose web server	172.16.0.111
		Subnet-A02	172.16.1.0/24	rtb-VPC-A	ECS-A02		172.16.1.91

VPC Name	VPC CIDR Block	Subnet Name	Subnet CIDR Block	Subnet Route Table	ECS Name	Security Group	Private IP Address
VPC-B	10.0.0.0/16	Subnet-B01	10.0.0.0/24	rtb-VPC-B	ECS-B01		10.0.0.139
VPC-C	192.168.0.0/16	Subnet-C01	192.168.0.0/24	rtb-VPC-C	ECS-C01		192.168.0.194
VPC-D	10.2.0.0/16	Subnet-D01	10.2.0.0/24	rtb-VPC-D	ECS-D01		10.2.0.237
VPC-E	10.3.0.0/16	Subnet-E01	10.3.0.0/24	rtb-VPC-E	ECS-E01		10.3.0.87
VPC-F	172.17.0.0/16	Subnet-F01	172.17.0.0/24	rtb-VPC-F	ECS-F01		172.17.0.103
VPC-G	10.4.0.0/16	Subnet-G01	10.4.0.0/24	rtb-VPC-G	ECS-G01		10.4.0.10

Table 8-28 Peering relationships (IPv4)

Peering Relationship	Peering Connection Name	Local VPC	Peer VPC
VPC-A is peered with VPC-B.	Peering-AB	VPC-A	VPC-B
VPC-A is peered with VPC-C.	Peering-AC	VPC-A	VPC-C
VPC-A is peered with VPC-D.	Peering-AD	VPC-A	VPC-D
VPC-A is peered with VPC-E.	Peering-AE	VPC-A	VPC-E
VPC-A is peered with VPC-F.	Peering-AF	VPC-A	VPC-F
VPC-A is peered with VPC-G.	Peering-AG	VPC-A	VPC-G

After the VPC peering connections are created, add the following routes to the route tables of the local and peer VPCs:

Table 8-29 VPC route table details (IPv4)

Route Table	Destination	Next Hop	Route Type	Description
rtb-VPC-A	172.16.0.0/24	Local	System	Local routes are automatically added for communications within a VPC.
	172.16.1.0/24	Local	System	
	10.0.0.0/16 (VPC-B)	Peering-AB	Custom	Add a route with the CIDR block of VPC-B as the destination and Peering-AB as the next hop.
	192.168.0.0/16 (VPC-C)	Peering-AC	Custom	Add a route with the CIDR block of VPC-C as the destination and Peering-AC as the next hop.
	10.2.0.0/16 (VPC-D)	Peering-AD	Custom	Add a route with the CIDR block of VPC-D as the destination and Peering-AD as the next hop.
	10.3.0.0/16 (VPC-E)	Peering-AE	Custom	Add a route with the CIDR block of VPC-E as the destination and Peering-AE as the next hop.
	172.17.0.0/16 (VPC-F)	Peering-AF	Custom	Add a route with the CIDR block of VPC-F as the destination and Peering-AF as the next hop.
	10.4.0.0/16 (VPC-G)	Peering-AG	Custom	Add a route with the CIDR block of VPC-G as the destination and Peering-AG as the next hop.
rtb-VPC-B	10.0.0.0/24	Local	System	Local routes are automatically added for communications within a VPC.
	172.16.0.0/16 (VPC-A)	Peering-AB	Custom	Add a route with the CIDR block of VPC-A as the destination and Peering-AB as the next hop.
rtb-VPC-C	192.168.0.0/24	Local	System	Local routes are automatically added for communications within a VPC.
	172.16.0.0/16 (VPC-A)	Peering-AC	Custom	Add a route with the CIDR block of VPC-A as the destination and Peering-AC as the next hop.
rtb-VPC-D	10.2.0.0/24	Local	System	Local routes are automatically added for communications within a VPC.

Route Table	Destination	Next Hop	Route Type	Description
	172.16.0.0/16 (VPC-A)	Peering-AD	Custom	Add a route with the CIDR block of VPC-A as the destination and Peering-AD as the next hop.
rtb-VPC-E	10.3.0.0/24	Local	System	Local routes are automatically added for communications within a VPC.
	172.16.0.0/16 (VPC-A)	Peering-AE	Custom	Add a route with the CIDR block of VPC-A as the destination and Peering-AE as the next hop.
rtb-VPC-F	172.17.0.0/24	Local	System	Local routes are automatically added for communications within a VPC.
	172.16.0.0/16 (VPC-A)	Peering-AF	Custom	Add a route with the CIDR block of VPC-A as the destination and Peering-AF as the next hop.
rtb-VPC-G	10.4.0.0/24	Local	System	Local routes are automatically added for communications within a VPC.
	172.16.0.0/16 (VPC-A)	Peering-AG	Custom	Add a route with the CIDR block of VPC-A as the destination and Peering-AG as the next hop.

NOTE

If the destination of a route in a route table of a VPC is set to the CIDR block of a peer VPC and the CIDR blocks of the two VPCs do not overlap, the VPCs can have full access to each other's resources.

One Central VPC Peered with Multiple VPCs (IPv6)

Create a VPC peering connection between VPC-A and VPC-B, between VPC-A and VPC-C, between VPC-A and VPC-D, between VPC-A and VPC-E, between VPC-A and VPC-F, and between VPC-A and VPC-G. Each VPC has IPv6 subnets. The IPv4 CIDR blocks of these VPCs do not overlap.

- For details about resource planning, see **Table 8-30**.
- For details about VPC peering relationships, see **Table 8-31**.

Figure 8-13 Networking diagram (IPv6)

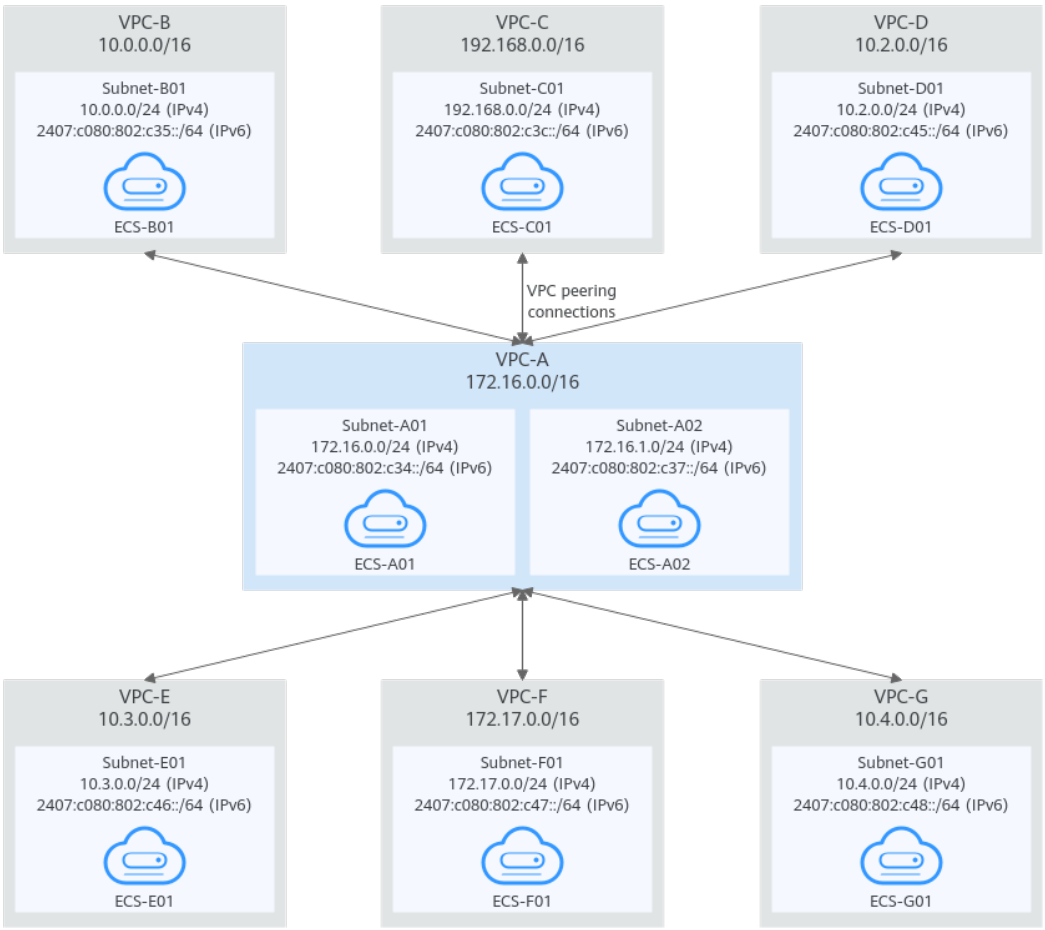


Table 8-30 Resource planning details (IPv6)

VPC Name	VPC CIDR Block	Subnet Name	Subnet CIDR Block	Subnet Route Table	ECS Name	Security Group	Private IP Address
VPC-A	172.16.0.0/16	Subnet-A01	• IPv4: 172.16.0.0/24 • IPv6: 2407:c080:802:c34::/64	rtb-VPC-A	ECS-A01	sg-web: general-purpose web server	• IPv4: 172.16.0.111 • IPv6: 2407:c080:802:c34:a925:f12e:cfa0:8edb

VPC Name	VPC CIDR Block	Subnet Name	Subnet CIDR Block	Subnet Route Table	ECS Name	Security Group	Private IP Address
		Subnet-A02	• IPv4: 172.16.1.0/24 • IPv6: 2407:c080:802:c37::/64	rtb-VPC-A	ECS-A02		• IPv4: 172.16.1.91 • IPv6: 2407:c080:802:c37:594b:4c0f:2fcd:8b72
VPC-B	10.0.0.0/16	Subnet-B01	• IPv4: 10.0.0.0/24 • IPv6: 2407:c080:802:c35::/64	rtb-VPC-B	ECS-B01		• IPv4: 10.0.0.139 • IPv6: 2407:c080:802:c35:493:33f4:4531:5162
VPC-C	192.168.0.0/16	Subnet-C01	• IPv4: 192.168.0.0/24 • IPv6: 2407:c080:802:c3c::/64	rtb-VPC-C	ECS-C01		• IPv4: 192.168.0.194 • IPv6: 2407:c080:802:c3c:d2f3:d891:24f5:f4af
VPC-D	10.2.0.0/16	Subnet-D01	• IPv4: 10.2.0.0/24 • IPv6: 2407:c080:802:c45::/64	rtb-VPC-D	ECS-D01		• IPv4: 10.2.0.237 • IPv6: 2407:c080:802:c45:6bb7:f161:3596:6e4c

VPC Name	VPC CIDR Block	Subnet Name	Subnet CIDR Block	Subnet Route Table	ECS Name	Security Group	Private IP Address
VPC-E	10.3.0.0/16	Subnet-E01	• IPv4: 10.3.0.0/24 • IPv6: 2407:c080:802:c46::/64	rtb-VPC-E	ECS-E01		• IPv4: 10.3.0.87 • IPv6: 2407:c080:802:c46:2a2f:558a:85da:4c70
VPC-F	172.17.0.0/16	Subnet-F01	• IPv4: 172.17.0.0/24 • IPv6: 2407:c080:802:c47::/64	rtb-VPC-F	ECS-F01		• IPv4: 172.17.0.103 • IPv6: 2407:c080:802:c47:b5e2:e6f0:c42b:44fd
VPC-G	10.4.0.0/16	Subnet-G01	• IPv4: 10.4.0.0/24 • IPv6: 2407:c080:802:c48::/64	rtb-VPC-G	ECS-G01		• IPv4: 10.4.0.10 • IPv6: 2407:c080:802:c48:3020:f48c:4e54:aa17

Table 8-31 Peering relationships (IPv6)

Peering Relationship	Peering Connection Name	Local VPC	Peer VPC
VPC-A is peered with VPC-B.	Peering-AB	VPC-A	VPC-B

Peering Relationship	Peering Connection Name	Local VPC	Peer VPC
VPC-A is peered with VPC-C.	Peering-AC	VPC-A	VPC-C
VPC-A is peered with VPC-D.	Peering-AD	VPC-A	VPC-D
VPC-A is peered with VPC-E.	Peering-AE	VPC-A	VPC-E
VPC-A is peered with VPC-F.	Peering-AF	VPC-A	VPC-F
VPC-A is peered with VPC-G.	Peering-AG	VPC-A	VPC-G

After the VPC peering connections are created, add the following routes to the route tables of the local and peer VPCs:

Table 8-32 VPC route table details (IPv6)

Route Table	Destination	Next Hop	Route Type	Description
rtb-VPC-A	172.16.0.0/24	Local	System	Local routes are automatically added for communications within a VPC.
	2407:c080:802:c34::/64	Local	System	
	172.16.1.0/24	Local	System	
	2407:c080:802:c37::/64	Local	System	
	10.0.0.0/16 (VPC-B)	Peering-AB	Custom	Add a route with the CIDR block of VPC-B as the destination and Peering-AB as the next hop for IPv4 communication.
	2407:c080:802:c35::/64 (Subnet-B01)	Peering-AB	Custom	Add a route with the IPv6 CIDR block of Subnet-B01 as the destination and Peering-AB as the next hop for IPv6 communication.

Route Table	Destination	Next Hop	Route Type	Description
	192.168.0.0/16 (VPC-C)	Peering-AC	Custom	Add a route with the CIDR block of VPC-C as the destination and Peering-AC as the next hop for IPv4 communication.
	2407:c080:802:c3c::/64 (Subnet-C01)	Peering-AC	Custom	Add a route with the IPv6 CIDR block of Subnet-C01 as the destination and Peering-AC as the next hop for IPv6 communication.
	10.2.0.0/16 (VPC-D)	Peering-AD	Custom	Add a route with the CIDR block of VPC-D as the destination and Peering-AD as the next hop for IPv4 communication.
	2407:c080:802:c45::/64 (Subnet-D01)	Peering-AD	Custom	Add a route with the IPv6 CIDR block of Subnet-D01 as the destination and Peering-AD as the next hop for IPv6 communication.
	10.3.0.0/16 (VPC-E)	Peering-AE	Custom	Add a route with the CIDR block of VPC-E as the destination and Peering-AE as the next hop for IPv4 communication.
	2407:c080:802:c46::/64 (Subnet-E01)	Peering-AE	Custom	Add a route with the IPv6 CIDR block of Subnet-E01 as the destination and Peering-AE as the next hop for IPv6 communication.
	172.17.0.0/16 (VPC-F)	Peering-AF	Custom	Add a route with the CIDR block of VPC-F as the destination and Peering-AF as the next hop for IPv4 communication.
	2407:c080:802:c47::/64 (Subnet-F01)	Peering-AF	Custom	Add a route with the IPv6 CIDR block of Subnet-F01 as the destination and Peering-AF as the next hop for IPv6 communication.
	10.4.0.0/16 (VPC-G)	Peering-AG	Custom	Add a route with the CIDR block of VPC-G as the destination and Peering-AG as the next hop for IPv4 communication.
	2407:c080:802:c48::/64 (Subnet-G01)	Peering-AG	Custom	Add a route with the IPv6 CIDR block of Subnet-G01 as the destination and Peering-AG as the next hop for IPv6 communication.

Route Table	Destination	Next Hop	Route Type	Description
rtb-VPC-B	10.0.0.0/24	Local	System	Local routes are automatically added for communications within a VPC.
	2407:c080:802:c35::/64	Local	System	
	172.16.0.0/16 (VPC-A)	Peering-AB	Custom	Add a route with the CIDR block of VPC-A as the destination and Peering-AB as the next hop for IPv4 communication.
	2407:c080:802:c34::/64 (Subnet-A01)	Peering-AB	Custom	Add routes with the IPv6 CIDR blocks of Subnet-A01 and Subnet-A02 as the destinations and Peering-AB as the next hop for IPv6 communication.
	2407:c080:802:c37::/64 (Subnet-A02)	Peering-AB	Custom	
rtb-VPC-C	192.168.0.0/24	Local	System	Local routes are automatically added for communications within a VPC.
	2407:c080:802:c3c::/64	Local	System	
	172.16.0.0/16 (VPC-A)	Peering-AC	Custom	Add a route with the CIDR block of VPC-A as the destination and Peering-AC as the next hop for IPv4 communication.
	2407:c080:802:c34::/64 (Subnet-A01)	Peering-AC	Custom	Add routes with the IPv6 CIDR blocks of Subnet-A01 and Subnet-A02 as the destinations and Peering-AC as the next hop for IPv6 communication.
	2407:c080:802:c37::/64 (Subnet-A02)	Peering-AC	Custom	
rtb-VPC-D	10.2.0.0/24	Local	System	Local routes are automatically added for communications within a VPC.
	2407:c080:802:c45::/64	Local	System	
	172.16.0.0/16 (VPC-A)	Peering-AD	Custom	Add a route with the CIDR block of VPC-A as the destination and Peering-AD as the next hop for IPv4 communication.

Route Table	Destination	Next Hop	Route Type	Description
	2407:c080:802:c34::/64 (Subnet-A01)	Peering-AD	Custom	Add routes with the IPv6 CIDR blocks of Subnet-A01 and Subnet-A02 as the destinations and Peering-AD as the next hop for IPv6 communication.
	2407:c080:802:c37::/64 (Subnet-A02)	Peering-AD	Custom	
rtb-VPC-E	10.3.0.0/24	Local	System	Local routes are automatically added for communications within a VPC.
	2407:c080:802:c46::/64	Local	System	
	172.16.0.0/16 (VPC-A)	Peering-AE	Custom	Add a route with the CIDR block of VPC-A as the destination and Peering-AE as the next hop.
	2407:c080:802:c34::/64 (Subnet-A01)	Peering-AE	Custom	Add routes with the IPv6 CIDR blocks of Subnet-A01 and Subnet-A02 as the destinations and Peering-AE as the next hop for IPv6 communication.
	2407:c080:802:c37::/64 (Subnet-A02)	Peering-AE	Custom	
rtb-VPC-F	172.17.0.0/24	Local	System	Local routes are automatically added for communications within a VPC.
	2407:c080:802:c47::/64	Local	System	
	172.16.0.0/16 (VPC-A)	Peering-AF	Custom	Add a route with the CIDR block of VPC-A as the destination and Peering-AF as the next hop.
	2407:c080:802:c34::/64 (Subnet-A01)	Peering-AF	Custom	Add routes with the IPv6 CIDR blocks of Subnet-A01 and Subnet-A02 as the destinations and Peering-AF as the next hop for IPv6 communication.
	2407:c080:802:c37::/64 (Subnet-A02)	Peering-AF	Custom	
rtb-VPC-G	10.4.0.0/24	Local	System	Local routes are automatically added for communications within a VPC.
	2407:c080:802:c48::/64	Local	System	

Route Table	Destination	Next Hop	Route Type	Description
	172.16.0.0/16 (VPC-A)	Peering-AG	Custom	Add a route with the CIDR block of VPC-A as the destination and Peering-AG as the next hop.
	2407:c080:802:c34::/64 (Subnet-A01)	Peering-AG	Custom	Add routes with the IPv6 CIDR blocks of Subnet-A01 and Subnet-A02 as the destinations and Peering-AG as the next hop for IPv6 communication.
	2407:c080:802:c37::/64 (Subnet-A02)	Peering-AG	Custom	

NOTE

You can view IPv6 addresses of VPC subnets on the management console. To enable communication between entire CIDR blocks of two VPCs, you need to add routes with IPv6 CIDR blocks of all subnets in the VPCs as the destinations one by one.

8.2.3 Using a VPC Peering Connection to Connect Subnets in Two VPCs

You can configure a VPC peering connection and set the destination of the routes added to VPC route tables to the subnet CIDR block of the peer VPC. In this way, all resources in the VPC subnets are connected. **Table 8-33** shows example scenarios.

Table 8-33 Scenario description

Scenario	Scenario Description	IP Address Version	Example
Two VPCs peered to two subnets in a central VPC	You have a central VPC that requires access to the multiple other VPCs. The other VPCs need to be isolated from each other. • The central VPC has separate sets of resources in different subnets. • The other VPCs require access to some of the resources, but not all of them.	IPv4	**Two VPCs Peered to Two Subnets in a Central VPC (IPv4)**
		IPv6/IPv4	**Two VPCs Peered to Two Subnets in a Central VPC (IPv6/IPv4)**

Scenario	Scenario Description	IP Address Version	Example
One central VPC peered to specific subnets in two VPCs	You have a central VPC that requires access to two other VPCs. The other VPCs need to be isolated from each other. • The central VPC has public resources deployed and the other VPCs require access to all resources in the central VPC. • Other VPCs have multiple subnets and only one in each VPC is used for accessing resources in the central VPC.	IPv4	**One Central VPC Peered to Specific Subnets in Two VPCs (IPv4)**
One central VPC peered to overlapping subnets from two VPCs	This scenario is similar to the preceding one. If two VPCs with overlapping subnets need to peer with the central VPC, traffic may fail to be forwarded to the required destination. To prevent this, plan the network according to this example.	IPv4	**One Central VPC Peered to Overlapping Subnets from Two VPCs (IPv4)**

Two VPCs Peered to Two Subnets in a Central VPC (IPv4)

The central VPC-A has two subnets, Subnet-A01 and Subnet-A02. The subnets are associated with different route tables. You need to create Peering-AB between Subnet-A01 and VPC-B, and Peering-AC between Subnet-A02 and VPC-C. VPC-B and VPC-C have the same CIDR block. However, there will be no route conflicts because the two subnets in VPC-A are associated with different route tables.

- For details about resource planning, see **Table 8-34**.
- For details about VPC peering relationships, see **Table 8-35**.

Figure 8-14 Networking diagram (IPv4)

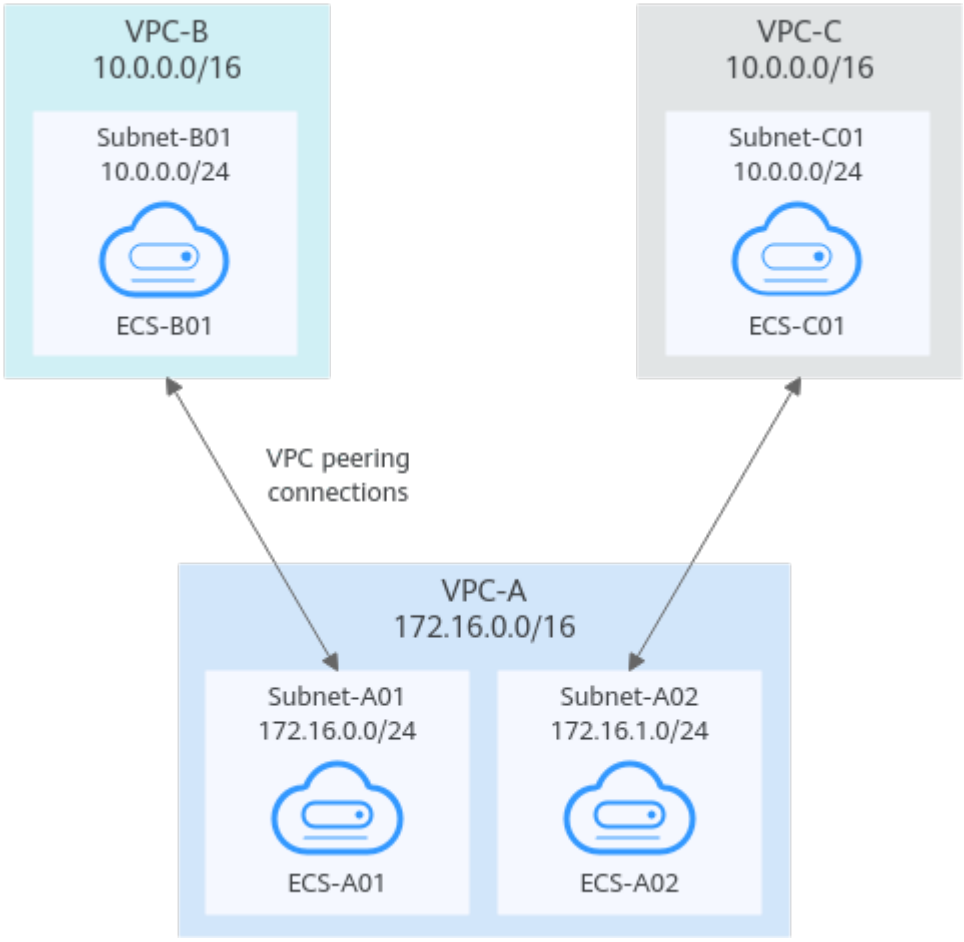


Table 8-34 Resource planning details (IPv4)

VPC Name	VPC CIDR Block	Subnet Name	Subnet CIDR Block	VPC Route Table	ECS Name	Security Group	Private IP Address
VPC-A	172.16.0.0/16	Subnet-A01	172.16.0.0/24	rtb-VPC-A01	ECS-A01	sg-web: general-purpose web server	172.16.0.111
		Subnet-A02	172.16.1.0/24	rtb-VPC-A02	ECS-A02		172.16.1.91
VPC-B	10.0.0.0/16	Subnet-B01	10.0.0.0/24	rtb-VPC-B	ECS-B01		10.0.0.139
VPC-C	10.0.0.0/16	Subnet-C01	10.0.0.0/24	rtb-VPC-C	ECS-C01		10.0.0.71

NOTE

VPC-A has two route tables. Route table rtb-VPC-A01 is associated with Subnet-A01, and route table rtb-VPC-A02 is associated with Subnet-A02. The two subnets can communicate with each other.

Table 8-35 Peering relationships (IPv4)

Peering Relationship	Peering Connection Name	Local VPC	Peer VPC
Subnet-A01 of VPC-A is peered to VPC-B.	Peering-AB	VPC-A	VPC-B
Subnet-A02 of VPC-A is peered to VPC-C.	Peering-AC	VPC-A	VPC-C

After the VPC peering connections are created, add the following routes to the route tables of the local and peer VPCs:

Table 8-36 VPC route table details (IPv4)

Route Table	Destination	Next Hop	Route Type	Description
rtb-VPC-A01	172.16.0.0/24	Local	System	Local routes are automatically added for communications within a VPC.
	172.16.1.0/24	Local	System	
	10.0.0.0/16 (VPC-B)	Peering-AB	Custom	Add a route with the CIDR block of VPC-B as the destination and Peering-AB as the next hop.
rtb-VPC-A02	172.16.0.0/24	Local	System	Local routes are automatically added for communications within a VPC.
	172.16.1.0/24	Local	System	
	10.0.0.0/16 (VPC-C)	Peering-AC	Custom	Add a route with the CIDR block of VPC-C as the destination and Peering-AC as the next hop.
rtb-VPC-B	10.0.0.0/24	Local	System	Local routes are automatically added for communications within a VPC.

Route Table	Destination	Next Hop	Route Type	Description
	172.16.0.0/24 (Subnet-A01)	Peering-AB	Custom	Add a route with the CIDR block of Subnet-A01 as the destination and Peering-AB as the next hop.
rtb-VPC-C	10.0.0.0/24	Local	System	Local routes are automatically added for communications within a VPC.
	172.16.1.0/24 (Subnet-A02)	Peering-AC	Custom	Add a route with the CIDR block of Subnet-A02 as the destination and Peering-AC as the next hop.

Two VPCs Peered to Two Subnets in a Central VPC (IPv6/IPv4)

The central VPC-A has two subnets, Subnet-A01 and Subnet-A02. The subnets are associated with different route tables. You need to create Peering-AB between Subnet-A01 and VPC-B for IPv6 communication, and Peering-AC between Subnet-A02 and VPC-C for IPv4 communication. VPC-B and VPC-C have the same CIDR block. However, there will be no route conflicts because the two subnets in VPC-A are associated with different route tables.

- For details about resource planning, see **Table 8-37**.
- For details about VPC peering relationships, see **Table 8-38**.

Figure 8-15 Networking diagram (IPv6/IPv4)

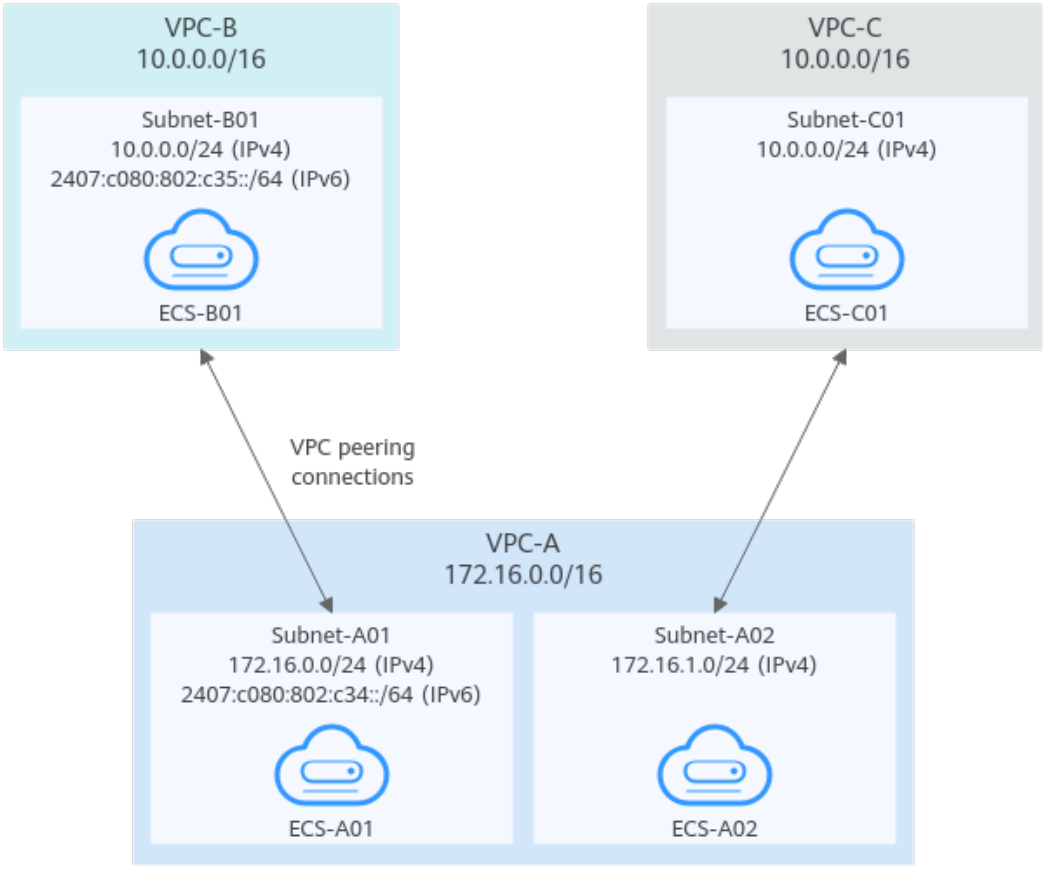


Table 8-37 Resource planning details (IPv6/IPv4)

VPC Name	VPC CIDR Block	Subnet Name	Subnet CIDR Block	VPC Route Table	ECS Name	Security Group	Private IP Address
VPC-A	172.16.0.0/16	Subnet-A01	• IPv4: 172.16.0.0/24 • IPv6: 2407:c080:802:c34::/64	rtb-VPC-A01	ECS-A01	sg-web: general-purpose web server	• IPv4: 172.16.0.111 • IPv6: 2407:c080:802:c34:a925:f12e:cfa0:8edb
		Subnet-A02	172.16.1.0/24	rtb-VPC-A02	ECS-A02		172.16.1.91

VPC Name	VPC CIDR Block	Subnet Name	Subnet CIDR Block	VPC Route Table	ECS Name	Security Group	Private IP Address
VPC-B	10.0.0.0/16	Subnet-B01	• IPv4: 10.0.0.0/24 • IPv6: 2407:c080:802:c35::/64	rtb-VPC-B	ECS-B01		• IPv4: 10.0.0.139 • IPv6: 2407:c080:802:c35:493:33f4:4531:5162
VPC-C	10.0.0.0/16	Subnet-C01	10.0.0.0/24	rtb-VPC-C	ECS-C01		10.0.0.71

NOTE

VPC-A has two route tables. Route table rtb-VPC-A01 is associated with Subnet-A01, and route table rtb-VPC-A02 is associated with Subnet-A02. The two subnets can communicate with each other.

Table 8-38 Peering relationships (IPv6/IPv4)

Peering Relationship	Peering Connection Name	Local VPC	Peer VPC
Subnet-A01 of VPC-A is peered to VPC-B. (IPv6)	Peering-AB	VPC-A	VPC-B
Subnet-A02 of VPC-A is peered to VPC-C. (IPv4)	Peering-AC	VPC-A	VPC-C

After the VPC peering connections are created, add the following routes to the route tables of the local and peer VPCs:

Table 8-39 VPC route table details (IPv6/IPv4)

Route Table	Destination	Next Hop	Route Type	Description
rtb-VPC-A01	172.16.0.0/24	Local	System	Local routes are automatically added for communications within a VPC.
	2407:c080:802:c34::/64	Local	System	
	172.16.1.0/24	Local	System	
	10.0.0.0/16 (VPC-B)	Peering-AB	Custom	Add a route with the CIDR block of VPC-B as the destination and Peering-AB as the next hop for IPv4 communication.
	2407:c080:802:c35::/64 (Subnet-B01)	Peering-AB	Custom	Add a route with the IPv6 CIDR block of Subnet-B01 as the destination and Peering-AB as the next hop for IPv6 communication.
rtb-VPC-A02	172.16.0.0/24	Local	System	Local routes are automatically added for communications within a VPC.
	2407:c080:802:c34::/64	Local	System	
	172.16.1.0/24	Local	System	
	10.0.0.0/16 (VPC-C)	Peering-AC	Custom	Add a route with the CIDR block of VPC-C as the destination and Peering-AC as the next hop for IPv4 communication.
rtb-VPC-B	10.0.0.0/24	Local	System	Local routes are automatically added for communications within a VPC.
	2407:c080:802:c35::/64	Local	System	
	172.16.0.0/24 (Subnet-A01)	Peering-AB	Custom	Add a route with the CIDR block of Subnet-A01 as the destination and Peering-AB as the next hop for IPv4 communication.
	2407:c080:802:c34::/64 (Subnet-A01)	Peering-AB	Custom	Add a route with the IPv6 CIDR block of Subnet-A01 as the destination and Peering-AB as the next hop for IPv6 communication.

Route Table	Destination	Next Hop	Route Type	Description
rtb-VPC-C	10.0.0.0/24	Local	System	Local routes are automatically added for communications within a VPC.
	172.16.1.0/24 (Subnet-A02)	Peering-AC	Custom	Add a route with the CIDR block of Subnet-A02 as the destination and Peering-AC as the next hop for IPv4 communication.

One Central VPC Peered to Specific Subnets in Two VPCs (IPv4)

You need to create Peering-AB between central VPC-A and Subnet-B01 in VPC-B, and Peering-AC between central VPC-A and Subnet-C02 in VPC-C. VPC-B and VPC-C have the same CIDR block, but the CIDR blocks of Subnet-B01 and Subnet-C02 do not overlap. Therefore, there will be no route conflicts.

- For details about resource planning, see **Table 8-40**.
- For details about VPC peering relationships, see **Table 8-41**.

Figure 8-16 Networking diagram (IPv4)

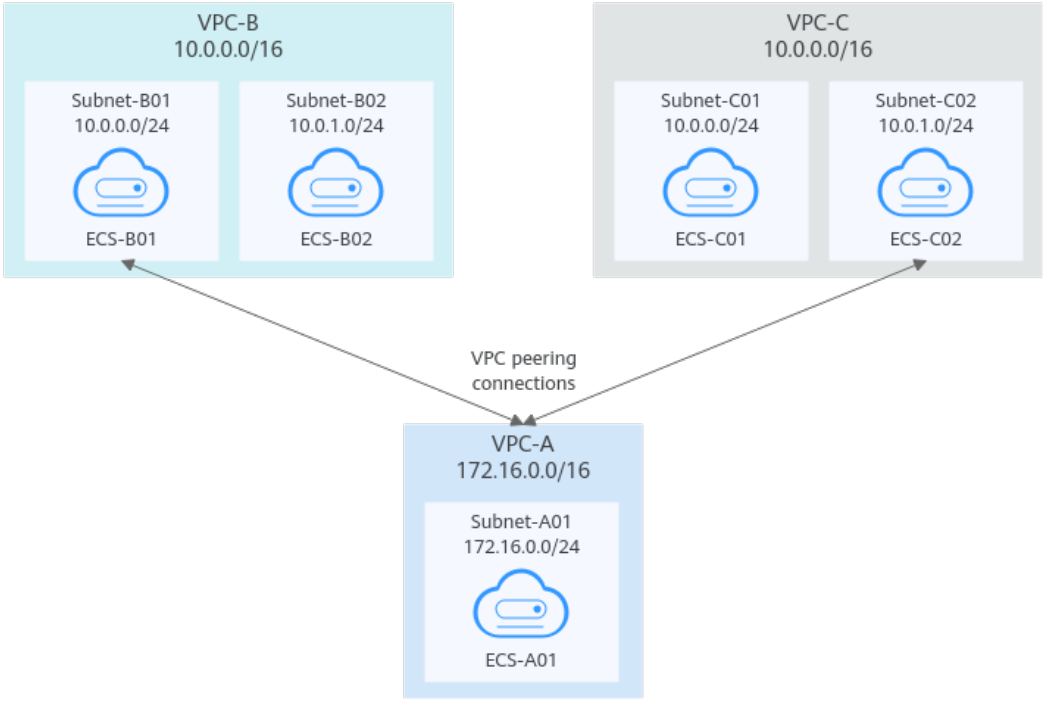


Table 8-40 Resource planning details (IPv4)

VPC Name	VPC CIDR Block	Subnet Name	Subnet CIDR Block	VPC Route Table	ECS Name	Security Group	Private IP Address
VPC-A	172.16.0.0/16	Subnet-A01	172.16.0.0/24	rtb-VPC-A	ECS-A01	sg-web: general-purpose web server	172.16.0.111
VPC-B	10.0.0.0/16	Subnet-B01	10.0.0.0/24	rtb-VPC-B	ECS-B01		10.0.0.139
		Subnet-B02	10.0.1.0/24	rtb-VPC-B	ECS-B02		10.0.1.167
VPC-C	10.0.0.0/16	Subnet-C01	10.0.0.0/24	rtb-VPC-C	ECS-C01		10.0.0.71
		Subnet-C02	10.0.1.0/24	rtb-VPC-C	ECS-C02		10.0.1.116

Table 8-41 Peering relationships (IPv4)

Peering Relationship	Peering Connection Name	Local VPC	Peer VPC
VPC-A is peered to Subnet-B01 of VPC-B.	Peering-AB	VPC-A	VPC-B
VPC-A is peered to Subnet-C02 of VPC-C.	Peering-AC	VPC-A	VPC-C

After the VPC peering connections are created, add the following routes to the route tables of the local and peer VPCs:

Table 8-42 VPC route table details (IPv4)

Route Table	Destination	Next Hop	Route Type	Description
rtb-VPC-A	172.16.0.0/24	Local	System	Local routes are automatically added for communications within a VPC.

Route Table	Destination	Next Hop	Route Type	Description
	10.0.0.0/24 (Subnet-B01)	Peering-AB	Custom	Add a route with the CIDR block of Subnet-B01 as the destination and Peering-AB as the next hop.
	10.0.1.0/24 (Subnet-C02)	Peering-AC	Custom	Add a route with the CIDR block of Subnet-C02 as the destination and Peering-AC as the next hop.
rtb-VPC-B	10.0.0.0/24	Local	System	Local routes are automatically added for communications within a VPC.
	10.0.1.0/24	Local	System	
	172.16.0.0/16 (VPC-A)	Peering-AB	Custom	Add a route with the CIDR block of VPC-A as the destination and Peering-AB as the next hop.
rtb-VPC-C	10.0.0.0/24	Local	System	Local routes are automatically added for communications within a VPC.
	10.0.1.0/24	Local	System	
	172.16.0.0/16 (VPC-A)	Peering-AC	Custom	Add a route with the CIDR block of VPC-A as the destination and Peering-AC as the next hop.

One Central VPC Peered to Overlapping Subnets from Two VPCs (IPv4)

If you want to create VPC peering connections between a VPC and multiple overlapping subnets from different VPCs, ensure that the destinations of the routes added for the peering connections do not conflict and traffic can be correctly forwarded.

In this example, you need to create Peering-AB between central VPC-A and Subnet-B02 in VPC-B, and Peering-AC between central VPC-A and Subnet-C02 in VPC-C. Subnet-B02 and Subnet-C02 have the same CIDR block, and ECS-B02 and ECS-C02 have the same private IP address (10.0.1.167/32).

- For details about resource planning, see **Table 8-43**.
- For details about VPC peering relationships, see **Table 8-44**.

Figure 8-17 Networking diagram (IPv4)

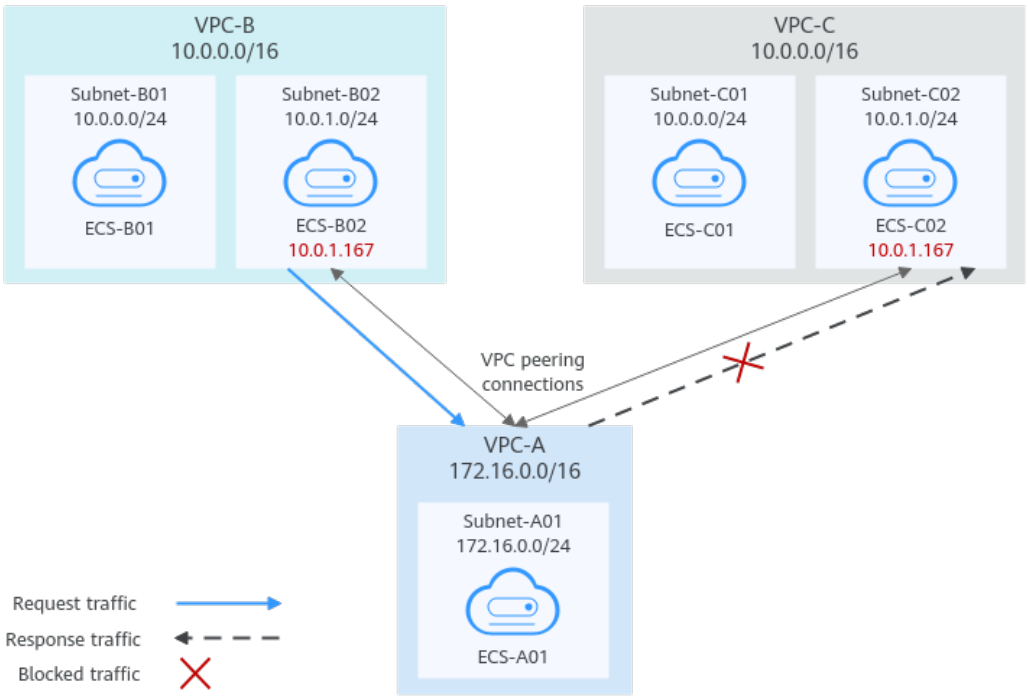


Table 8-43 Resource planning details (IPv4)

VPC Name	VPC CIDR Block	Subnet Name	Subnet CIDR Block	VPC Route Table	ECS Name	Security Group	Private IP Address
VPC-A	172.16.0.0/16	Subnet-A01	172.16.0.0/24	rtb-VPC-A	ECS-A01	sg-web: general-purpose web server	172.16.0.111
VPC-B	10.0.0.0/16	Subnet-B01	10.0.0.0/24	rtb-VPC-B	ECS-B01		10.0.0.139
		Subnet-B02	10.0.1.0/24	rtb-VPC-B	ECS-B02		10.0.1.167
VPC-C	10.0.0.0/16	Subnet-C01	10.0.0.0/24	rtb-VPC-C	ECS-C01		10.0.0.71
		Subnet-C02	10.0.1.0/24	rtb-VPC-C	ECS-C02		10.0.1.167

Table 8-44 Peering relationships (IPv4)

Peering Relationship	Peering Connection Name	Local VPC	Peer VPC
VPC-A is peered to Subnet-B02 of VPC-B.	Peering-AB	VPC-A	VPC-B
VPC-A is peered to Subnet-C02 of VPC-C.	Peering-AC	VPC-A	VPC-C

If you add routes to the route tables of the local and peer VPCs according to **Table 8-45**, the response traffic cannot be correctly forwarded. The details are as follows:

1. ECS-B02 in Subnet-B02 of VPC-B sends request traffic to VPC-A through the route with Peering-AB as the next hop in the rtb-VPC-B route table.
2. VPC-A receives the request traffic from ECS-B02 and expects to send the response traffic to ECS-B02. The rtb-VPC-A route table has the route with 10.0.1.167/32 as the destination, but its next hop is Peering-AC. The response traffic is incorrectly sent to VPC-C.
3. ECS-C02 in Subnet-C02 of VPC-C has the same private IP address (10.0.1.167/32) as ECS-B02. The response traffic is incorrectly sent to ECS-C02, and ECS-B02 cannot receive the response traffic.

Table 8-45 VPC route table details (IPv4)

Route Table	Destination	Next Hop	Route Type	Description
rtb-VPC-A	172.16.0.0/24	Local	System	Local routes are automatically added for communications within a VPC.
	10.0.1.0/24 (Subnet-C02)	Peering-AC	Custom	Add a route with the CIDR block of Subnet-C02 as the destination and Peering-AC as the next hop.
rtb-VPC-B	10.0.0.0/24	Local	System	Local routes are automatically added for communications within a VPC.
	10.0.1.0/24	Local	System	
	172.16.0.0/16 (VPC-A)	Peering-AB	Custom	Add a route with the CIDR block of VPC-A as the destination and Peering-AB as the next hop.

Route Table	Destination	Next Hop	Route Type	Description
rtb-VPC-C	10.0.0.0/24	Local	System	Local routes are automatically added for communications within a VPC.
	10.0.1.0/24	Local	System	
	172.16.0.0/16 (VPC-A)	Peering-AC	Custom	Add a route with the CIDR block of VPC-A as the destination and Peering-AC as the next hop.

If there are overlapping subnets, configure routes as follows to prevent traffic from being incorrectly forwarded:

- Suggestion 1: In the rtb-VPC-A route table, add a route with Peering-AB as the next hop and the private IP address of ECS-B02 (10.0.1.167/32) as the destination. The route with 10.0.1.167/32 as the destination is preferentially matched based on the longest prefix match rule to ensure that VPC-A sends the response traffic to ECS-B02. For more configurations, see **Using a VPC Peering Connection to Connect ECSs in Two VPCs**.

  Table 8-46 VPC route table details

Route Table	Destination	Next Hop	Route Type	Description
rtb-VPC-A	172.16.0.0/24	Local	System	Local routes are automatically added for communications within a VPC.
	10.0.1.167/32 (ECS-B02)	Peering-AB	Custom	Add a route with the private IP address of ECS-B02 as the destination and Peering-AB as the next hop.
	10.0.1.0/24 (Subnet-C02)	Peering-AC	Custom	Add a route with the CIDR block of Subnet-C02 as the destination and Peering-AC as the next hop.

- Suggestion 2: In the rtb-VPC-A route table, change the destination of the route with Peering-AC as the next hop from Subnet-C02 to Subnet-C01. Add a route with Peering-AB as the next hop and Subnet-B02 as the destination to ensure that VPC-A can send the response traffic to Subnet-B02 in VPC-B.

Table 8-47 VPC route table details

Route Table	Destination	Next Hop	Route Type	Description
rtb-VPC-A	172.16.0.0/24	Local	System	Local routes are automatically added for communications within a VPC.
	10.0.1.0/24 (Subnet-B02)	Peering-AB	Custom	Add a route with the CIDR block of Subnet-B02 as the destination and Peering-AB as the next hop.
	10.0.0.0/24 (Subnet-C01)	Peering-AC	Custom	Add a route with the CIDR block of Subnet-C01 as the destination and Peering-AC as the next hop.

8.2.4 Using a VPC Peering Connection to Connect ECSs in Two VPCs

You can configure a VPC peering connection and set the destination of the routes added to VPC route tables to the private IP address of ECS in the peer VPC. In this way, the two ECS are connected.

To enable traffic forwarding among these ECSs, you need to add routes with private IP addresses of these ECSs as the destinations and a VPC peering connection as the next hop to VPC route tables. **Table 8-48** shows example scenarios.

Table 8-48 Scenario description

Scenario	Scenario Description	IP Address Version	Example
ECS in a central VPC peered to ECSs in two other VPCs	You want a central VPC to communicate with the other two VPCs. However, you do not want the other two VPCs to communicate with each other. The other two VPCs have the same CIDR block and also include subnets that overlap. To prevent route conflicts in the central VPC, you can configure VPC peering connections to connect to specific ECSs in the other two VPCs.	IPv4	**ECS in a Central VPC Peered to ECSs in Two Other VPCs (IPv4)**

Scenario	Scenario Description	IP Address Version	Example
A central VPC peered with two other VPCs using longest prefix match	This scenario is similar to the preceding one. In addition to peering specific ECSs, you can create the following VPC peering connections based on the longest prefix match rule: • Create a VPC peering connection between the central VPC and an ECS in VPC-B • Create a VPC peering connection between the central VPC and a subnet in VPC-C This configuration expands the communication scope.	IPv4	**A Central VPC Peered with Two Other VPCs Using Longest Prefix Match (IPv4)**

ECS in a Central VPC Peered to ECSs in Two Other VPCs (IPv4)

You want to create a VPC peering connection between VPC-A and VPC-B, and between VPC-A and VPC-C. VPC-B and VPC-C have matching CIDR blocks. You can set the destinations of routes to private IP addresses of specific ECSs to limit traffic to these ECSs. If the destination of a route is not properly planned, traffic cannot be correctly forwarded. For details, see **One Central VPC Peered to Overlapping Subnets from Two VPCs (IPv4)**.

In this example, you need to create Peering-AB between ECS-A01-1 in VPC-A and ECS-B01 in VPC-B, and Peering-AC between ECS-A01-2 in VPC-A and ECS-C01 in VPC-C. Subnet-B01 and Subnet-C01 have matching CIDR blocks. The private IP addresses of ECS-B01 and ECS-C01 must be different. Otherwise, there will be route conflicts because the route table of VPC-A will have routes with the same destination.

- For details about resource planning, see **Table 8-49**.
- For details about VPC peering relationships, see **Table 8-50**.

Figure 8-18 Networking diagram (IPv4)

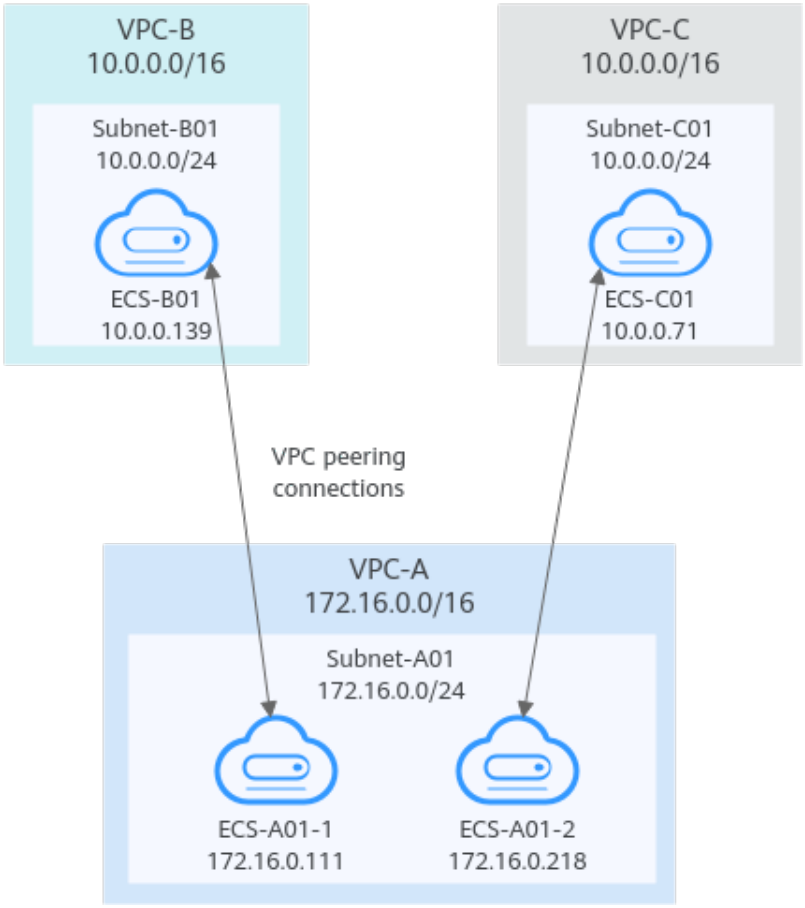


Table 8-49 Resource planning details (IPv4)

VPC Name	VPC CIDR Block	Subnet Name	Subnet CIDR Block	VPC Route Table	ECS Name	Security Group	Private IP Address
VPC-A	172.16.0.0/16	Subnet-A01	172.16.0.0/24	rtb-VPC-A	ECS-A01-1	sg-web: general-purpose web server	172.16.0.111
					ECS-A01-2		172.16.0.218
VPC-B	10.0.0.0/16	Subnet-B01	10.0.0.0/24	rtb-VPC-B	ECS-B01		10.0.0.139
VPC-C	10.0.0.0/16	Subnet-C01	10.0.0.0/24	rtb-VPC-C	ECS-C01		10.0.0.71

Table 8-50 Peering relationships (IPv4)

Peering Relationship	Peering Connection Name	Local VPC	Peer VPC
ECS-A01-1 in VPC-A is peered with ECS-B01 in VPC-B.	Peering-AB	VPC-A	VPC-B
ECS-A01-2 in VPC-A is peered with ECS-C01 in VPC-C.	Peering-AC	VPC-A	VPC-C

After the VPC peering connections are created, add the following routes to the route tables of the local and peer VPCs:

Table 8-51 VPC route table details (IPv4)

Route Table	Destination	Next Hop	Route Type	Description
rtb-VPC-A	172.16.0.0/24	Local	System	Local routes are automatically added for communications within a VPC.
	10.0.0.139/32 (ECS-B01)	Peering-AB	Custom	Add a route with the private IP address of ECS-B01 as the destination and Peering-AB as the next hop.
	10.0.0.71/32 (ECS-C01)	Peering-AC	Custom	Add a route with the private IP address of ECS-C01 as the destination and Peering-AC as the next hop.
rtb-VPC-B	10.0.0.0/24	Local	System	Local routes are automatically added for communications within a VPC.
	172.16.0.111/32 (ECS-A01-1)	Peering-AB	Custom	Add a route with the private IP address of ECS-A01-1 as the destination and Peering-AB as the next hop.
rtb-VPC-C	10.0.0.0/24	Local	System	Local routes are automatically added for communications within a VPC.

Route Table	Destination	Next Hop	Route Type	Description
	172.16.0.218/32 (ECS-A01-2)	Peering-AC	Custom	Add a route with the private IP address of ECS-A01-2 as the destination and Peering-AC as the next hop.

A Central VPC Peered with Two Other VPCs Using Longest Prefix Match (IPv4)

You want to create a VPC peering connection between VPC-A and VPC-B, and between VPC-A and VPC-C. VPC-B and VPC-C have matching CIDR blocks. You can set the destinations of routes to private IP addresses of specific ECSs to limit traffic to these ECSs. If the destination of a route is not properly planned, traffic cannot be correctly forwarded. For details, see **One Central VPC Peered to Overlapping Subnets from Two VPCs (IPv4)**.

In this example, you need to create Peering-AB between central VPC-A and ECS-B01 in VPC-B, and Peering-AC between central VPC-A and VPC-C. Subnet-B01 and Subnet-C01 have matching CIDR blocks. You can use the longest prefix match rule to control traffic forwarding.

- For details about resource planning, see **Table 8-52**.
- For details about VPC peering relationships, see **Table 8-53**.

Figure 8-19 Networking diagram (IPv4)

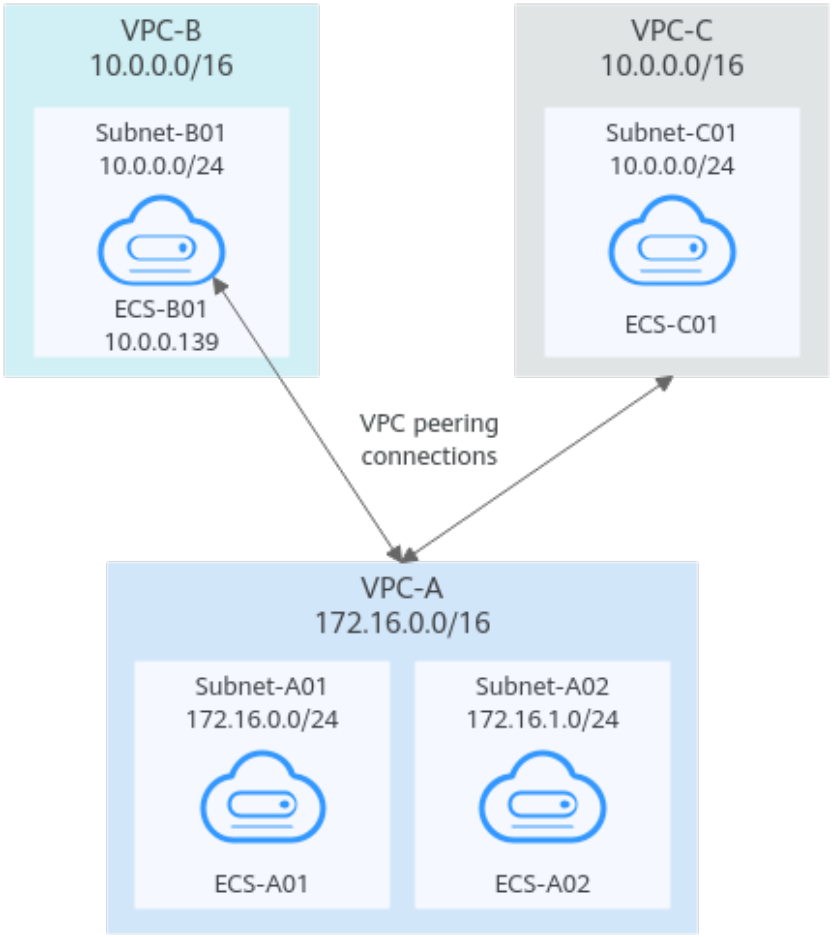


Table 8-52 Resource planning details (IPv4)

VPC Name	VPC CIDR Block	Subnet Name	Subnet CIDR Block	VPC Route Table	ECS Name	Security Group	Private IP Address
VPC-A	172.16.0.0/16	Subnet-A01	172.16.0.0/24	rtb-VPC-A	ECS-A01	sg-web: general-purpose web server	172.16.0.111
		Subnet-A02	172.16.1.0/24	rtb-VPC-A	ECS-A02		172.16.1.91
VPC-B	10.0.0.0/16	Subnet-B01	10.0.0.0/24	rtb-VPC-B	ECS-B01		10.0.0.139
VPC-C	10.0.0.0/16	Subnet-C01	10.0.0.0/24	rtb-VPC-C	ECS-C01		10.0.0.71

Table 8-53 Peering relationships (IPv4)

Peering Relationship	Peering Connection Name	Local VPC	Peer VPC
VPC-A is peered with ECS-B01 in VPC-B.	Peering-AB	VPC-A	VPC-B
VPC-A is peered with VPC-C.	Peering-AC	VPC-A	VPC-C

After the VPC peering connections are created, add the following routes to the route tables of the local and peer VPCs:

Table 8-54 VPC route table details (IPv4)

Route Table	Destination	Next Hop	Route Type	Description
rtb-VPC-A	172.16.0.0/24	Local	System	Local routes are automatically added for communications within a VPC.
	172.16.1.0/24	Local	System	
	10.0.0.139/32 (ECS-B01)	Peering-AB	Custom	Add a route with the private IP address of ECS-B01 as the destination and Peering-AB as the next hop.
	10.0.0.0/16 (VPC-C)	Peering-AC	Custom	Add a route with the CIDR block of VPC-C as the destination and Peering-AC as the next hop.
rtb-VPC-B	10.0.0.0/24	Local	System	Local routes are automatically added for communications within a VPC.
	172.16.0.0/16 (VPC-A)	Peering-AB	Custom	Add a route with the CIDR block of VPC-A as the destination and Peering-AB as the next hop.
rtb-VPC-C	10.0.0.0/24	Local	System	Local routes are automatically added for communications within a VPC.
	172.16.0.0/16 (VPC-A)	Peering-AC	Custom	Add a route with the CIDR block of VPC-A as the destination and Peering-AC as the next hop.

8.2.5 Unsupported VPC Peering Configurations

Scenarios

The VPC peering connection configurations are not supported in **Table 8-55**.

Table 8-55 Scenarios that VPC peering connections are invalid

Scenario	Example
• If VPCs with the same CIDR block also include subnets that overlap, VPC peering connections are not usable. • If two VPCs have overlapping CIDR blocks but some of their subnets do not overlap, you cannot create a VPC peering connection to connect specific subnets that do not overlap.	**Invalid VPC Peering for Overlapping VPC CIDR Blocks** • **VPCs with the same CIDR block also include subnets that overlap.** • **Two VPCs have overlapping CIDR blocks but some of their subnets do not overlap.**
VPC peering connections cannot enable ECSs in their VPCs to share an EIP to access the Internet. If VPC-A and VPC-B are peered and ECS-A01 in VPC-A has an EIP, ECS-B01 in VPC-B cannot access the Internet using the EIP bound to ECS-A01.	**Invalid VPC Peering for Sharing an EIP**

Invalid VPC Peering for Overlapping VPC CIDR Blocks

If two VPCs have overlapping CIDR blocks, the VPC peering connection may not take effect due to route conflicts. The following describes the reasons and configuration suggestions.

- VPCs with the same CIDR block also include subnets that overlap.

  VPC peering connections are not usable. As shown in **Table 8-56**, VPC-A and VPC-B, and their subnets have the same CIDR block. If you create a VPC peering connection between VPC-A and VPC-B, their route tables are shown in **Table 8-56**.

  In the rtb-VPC-A route table, the custom route for routing traffic from VPC-A to VPC-B and the local route have overlapping destinations. The local route has a higher priority and traffic will be forwarded within VPC-A and cannot reach VPC-B.

Figure 8-20 Networking diagram (IPv4)

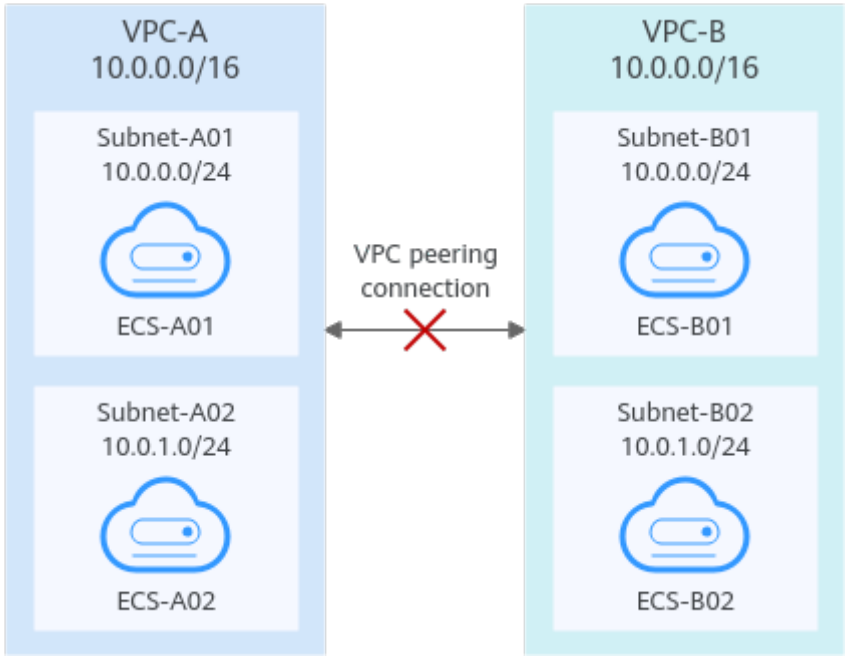
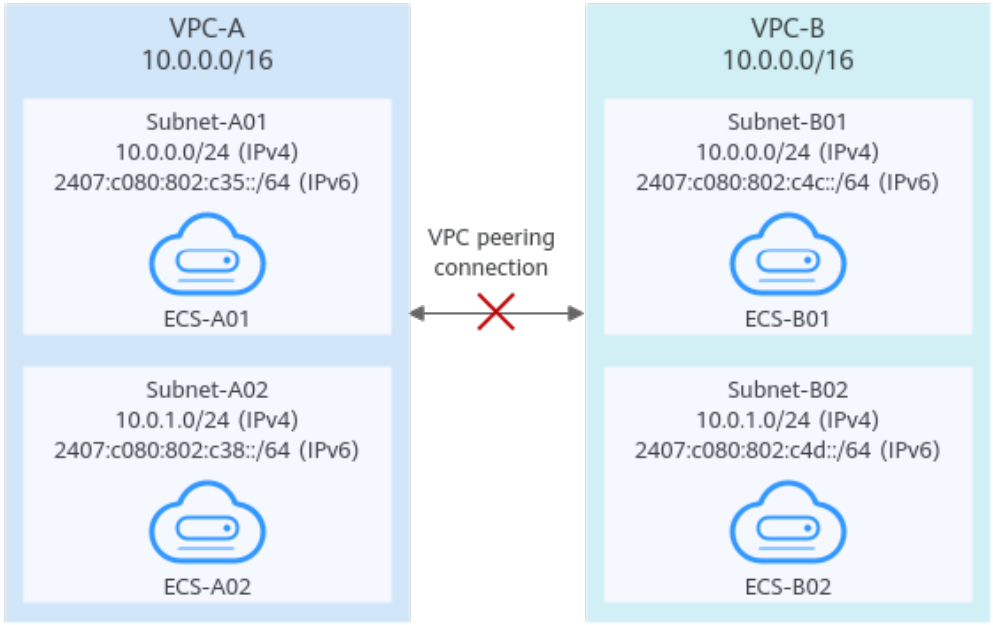


Table 8-56 VPC route table details

Route Table	Destination	Next Hop	Route Type	Description
rtb-VPC-A	10.0.0.0/24	Local	System	Local routes are automatically added for communications within a VPC.
	10.0.1.0/24	Local	System	
	10.0.0.0/16 (VPC-B)	Peering-AB	Custom	Add a route with the CIDR block of VPC-B as the destination and Peering-AB as the next hop.
rtb-VPC-B	10.0.0.0/24	Local	System	Local routes are automatically added for communications within a VPC.
	10.0.1.0/24	Local	System	
	10.0.0.0/16 (VPC-A)	Peering-AB	Custom	Add a route with the CIDR block of VPC-A as the destination and Peering-AB as the next hop.

If two VPCs want to use their IPv6 CIDR blocks for communication by a VPC peering connection but the IPv4 CIDR blocks of the VPCs or subnets overlap, the connection is not usable.

Figure 8-21 Networking diagram (IPv6)



- Two VPCs have overlapping CIDR blocks but some of their subnets do not overlap.

  VPC peering connections will not take effect in the following scenarios:

  – Connecting overlapping CIDR blocks of VPCs

    As shown in **Figure 8-22**, if you create a VPC peering connection between VPC-A and VPC-B, the VPC peering connection will not take effect because the two VPCs have the same CIDR block.

  – Connecting overlapping subnets from different VPCs

    If you create a VPC peering connection between Subnet-A01 and Subnet-B02, the route tables are shown in **Table 8-57**. In the rtb-VPC-B route table, the custom route for routing traffic from Subnet-B02 to Subnet-A01 and the local route have overlapping destinations. The local route has a higher priority and traffic will be forwarded within Subnet-B02 and cannot reach Subnet-A01.

Figure 8-22 Networking diagram (IPv4)

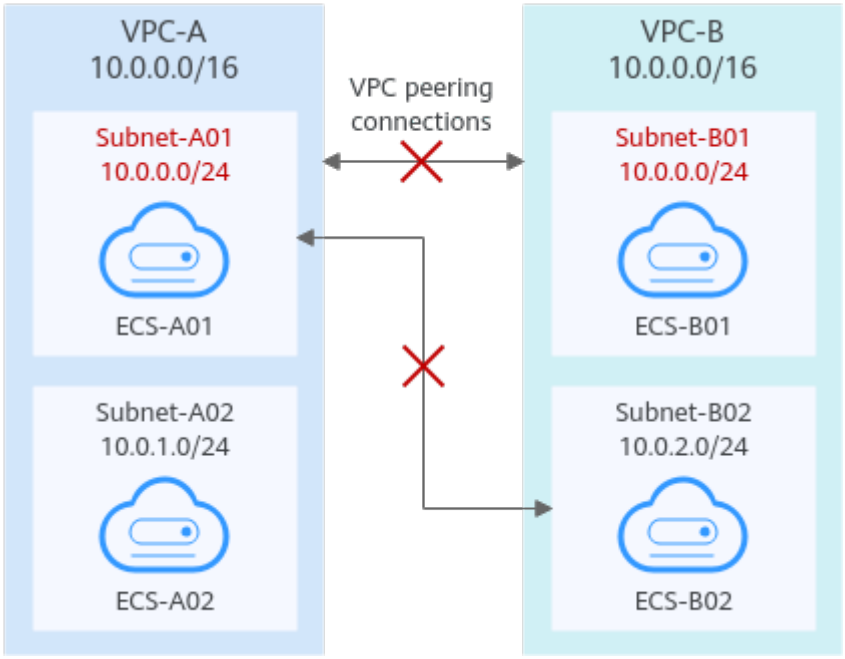
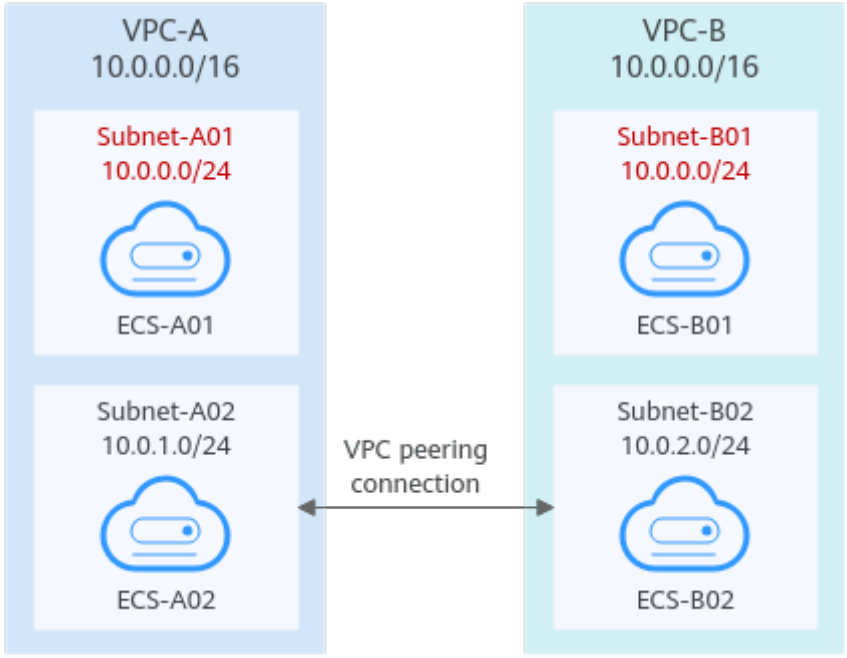


Table 8-57 VPC route table details

Route Table	Destination	Next Hop	Route Type	Description
rtb-VPC-A	10.0.0.0/24	Local	System	Local routes are automatically added for communications within a VPC.
	10.0.1.0/24	Local	System	
	10.0.2.0/24 (Subnet-B02)	Peering-AB	Custom	Add a route with the CIDR block of Subnet-B02 as the destination and Peering-AB as the next hop.
rtb-VPC-B	10.0.0.0/24	Local	System	Local routes are automatically added for communications within a VPC.
	10.0.2.0/24	Local	System	
	10.0.0.0/24 (Subnet-A01)	Peering-AB	Custom	Add a route with the CIDR block of Subnet-A01 as the destination and Peering-AB as the next hop.

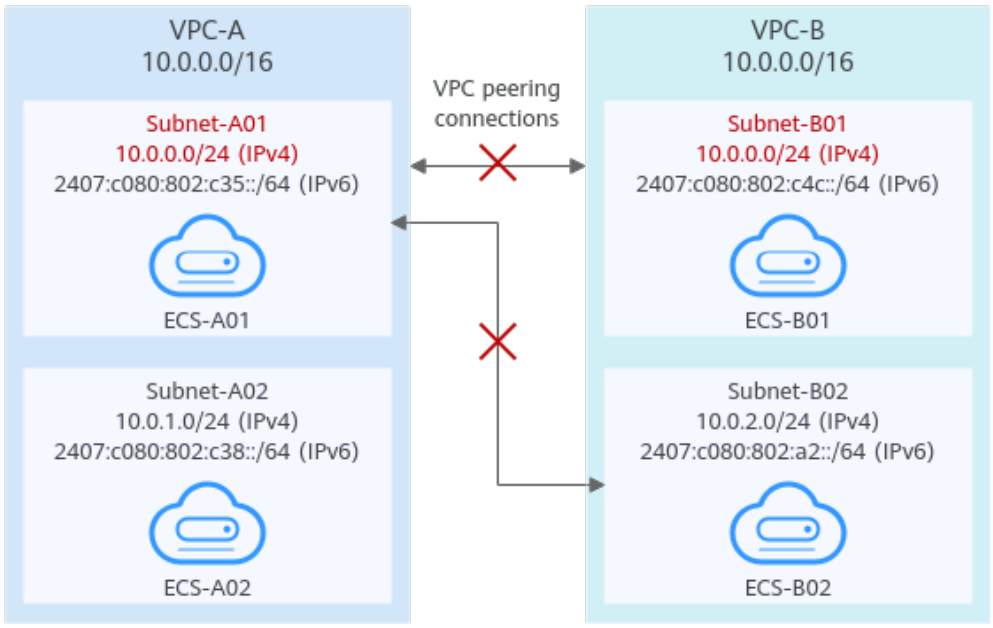
If the subnets connected by a VPC peering connection do not overlap, the connection will take effect. As shown in **Figure 8-23**, you can create a VPC peering connection between Subnet-A02 and Subnet-B02. In this case, the routes do not conflict and the VPC peering connection takes effect.

Figure 8-23 Networking diagram (IPv4)



If two VPCs want to use their IPv6 CIDR blocks for communication by a VPC peering connection but the IPv4 CIDR blocks of the VPCs or subnets overlap, the connection is not usable.

Figure 8-24 Networking diagram (IPv6)

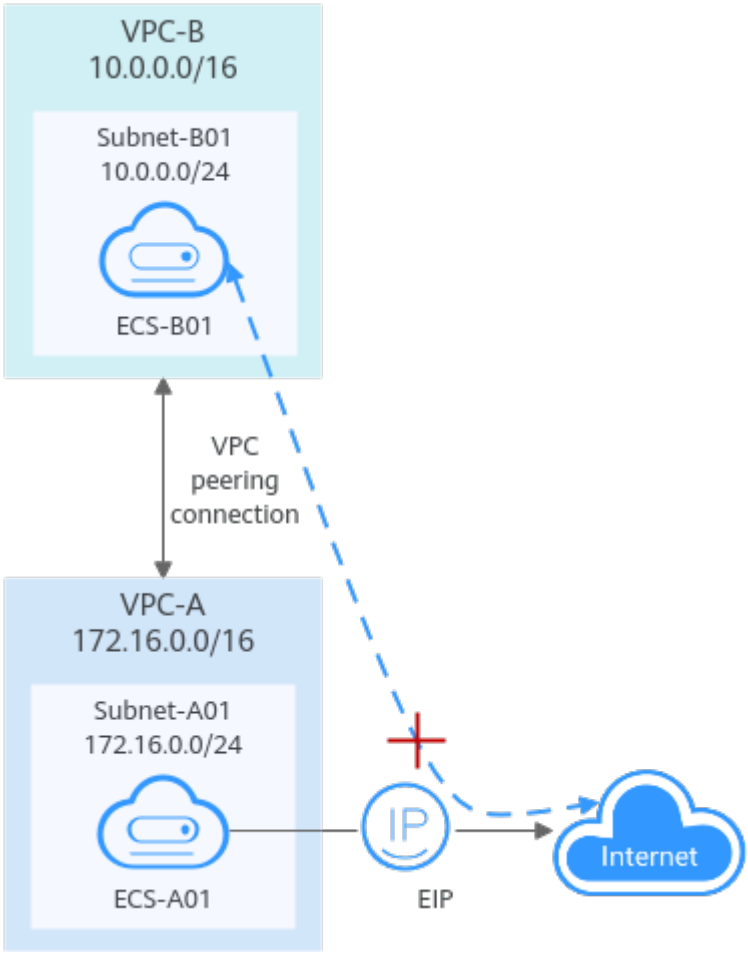


Invalid VPC Peering for Sharing an EIP

As shown in **Figure 8-25**, although VPC-A and VPC-B are peered and ECS-A01 in VPC-A has an EIP, ECS-B01 in VPC-B cannot access the Internet using the EIP bound to ECS-A01. If you want multiple resources to share an EIP, refer to **Using**

NAT Gateway and VPC Peering to Enable Communication Between VPCs and Internet.

Figure 8-25 Networking diagram



8.3 Creating a VPC Peering Connection to Connect Two VPCs in the Same Account

Scenarios

Two VPCs from the same region cannot communicate with each other by default, but you can use a VPC peering connection to connect them.

The following describes how to create a VPC peering connection to connect two VPCs (**vpc-A** and **vpc-B** in this example) in the same account. In this way, instances (the service server **ECS-A01** and database server **RDS-B01** in this example) in the two VPCs can communicate with each other.

The procedure is as follows:

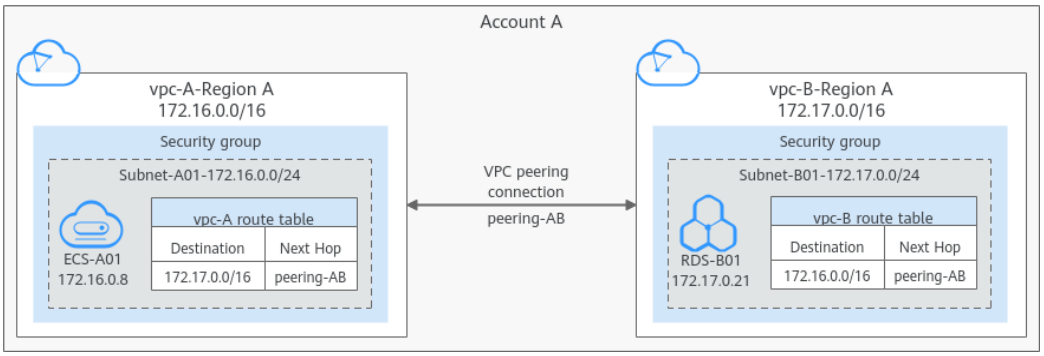
Step 1: Create a VPC Peering Connection

Step 2: Add Routes for the VPC Peering Connection

Step 3: Configure Security Group Rules for Instances in Local and Peer VPCs

Step 4: Verify Network Connectivity

Figure 8-26 Connecting two VPCs in an account using a VPC peering connection



NOTICE

Currently, VPC peering connections are free.

Notes and Constraints

- Only one VPC peering connection can be created between two VPCs at the same time.
- A VPC peering connection can only connect VPCs in the same region.
  - A VPC peering connection can enable a VPC created on the Huawei Cloud Chinese Mainland website to connect to one created on the Huawei Cloud International website, but the VPCs must be in the same region. For example, if the VPC created on the Chinese Mainland website is in the CN-Hong Kong region, then the VPC created on the International website must also be in the CN-Hong Kong region.
  - If you want to connect VPCs in different regions, you can use **Cloud Connect**.
  - If you only need a few ECSs in different regions to communicate with each other, you can **assign and bind EIPs to the ECSs**.
- If the local and peer VPCs have overlapping CIDR blocks, the VPC peering connection may not be usable.

  In this case, you can configure the network by referring to **VPC Peering Connection Usage Examples**.

Prerequisites

You have two VPCs from the same account in the same region. If you want to create one, see **Creating a VPC and Subnet**.

Step 1: Create a VPC Peering Connection

1. Go to the **VPC peering connection list page**.

2. In the upper right corner of the page, click **Create VPC Peering Connection**.
   The **Create VPC Peering Connection** page is displayed.
3. Configure the parameters as prompted.
   For details, see **Table 8-58**.

Figure 8-27 Creating a VPC peering connection

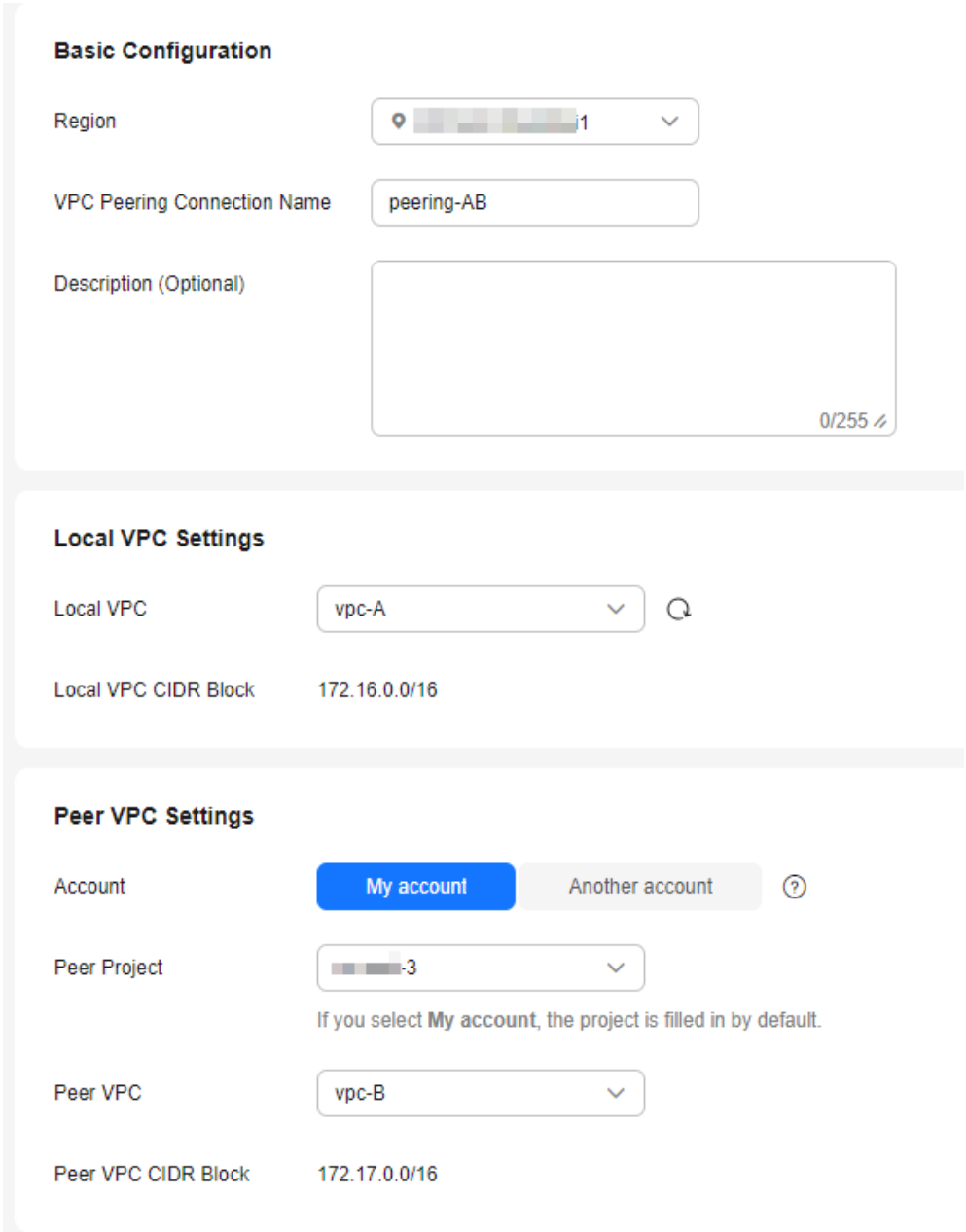


Table 8-58 Parameters for creating a VPC peering connection

Parameter	Description	Example Value
Region	Mandatory The region where the VPC peering connection is created. Select the region nearest to you to ensure the lowest latency possible.	CN-Hong Kong
VPC Peering Connection Name	Mandatory Enter a name for the VPC peering connection. The name can contain a maximum of 64 characters, including letters, digits, hyphens (-), and underscores (_).	peering-AB
Description (Optional)	Optional Enter a description of the VPC peering connection in the text box as required.	peering-AB connects **vpc-A** and **vpc-B**.
Local VPC	Mandatory VPC at one end of the VPC peering connection. You can select one from the drop-down list.	vpc-A
Local VPC CIDR Block	CIDR block of the selected local VPC	172.16.0.0/16
Account	Mandatory • Options: **My account** and **Another account** • Select **My account**.	My account
Peer Project	The project is selected in by default if **Account** is set to **My account**. In this example, **vpc-A** and **vpc-B** are created in region A, and the corresponding project of the account in region A is selected by default.	ab-cdef-1

Parameter	Description	Example Value
Peer VPC	This parameter is mandatory if **Account** is set to **My account**. VPC at the other end of the VPC peering connection. You can select one from the drop-down list.	vpc-B
Peer VPC CIDR Block	CIDR block of the selected peer VPC. If the local and peer VPCs have overlapping CIDR blocks, the VPC peering connection may not be usable. For details, see **VPC Peering Connection Usage Examples**.	172.17.0.0/16

4. Click **Create Now**.

   A dialog box for adding routes is displayed.

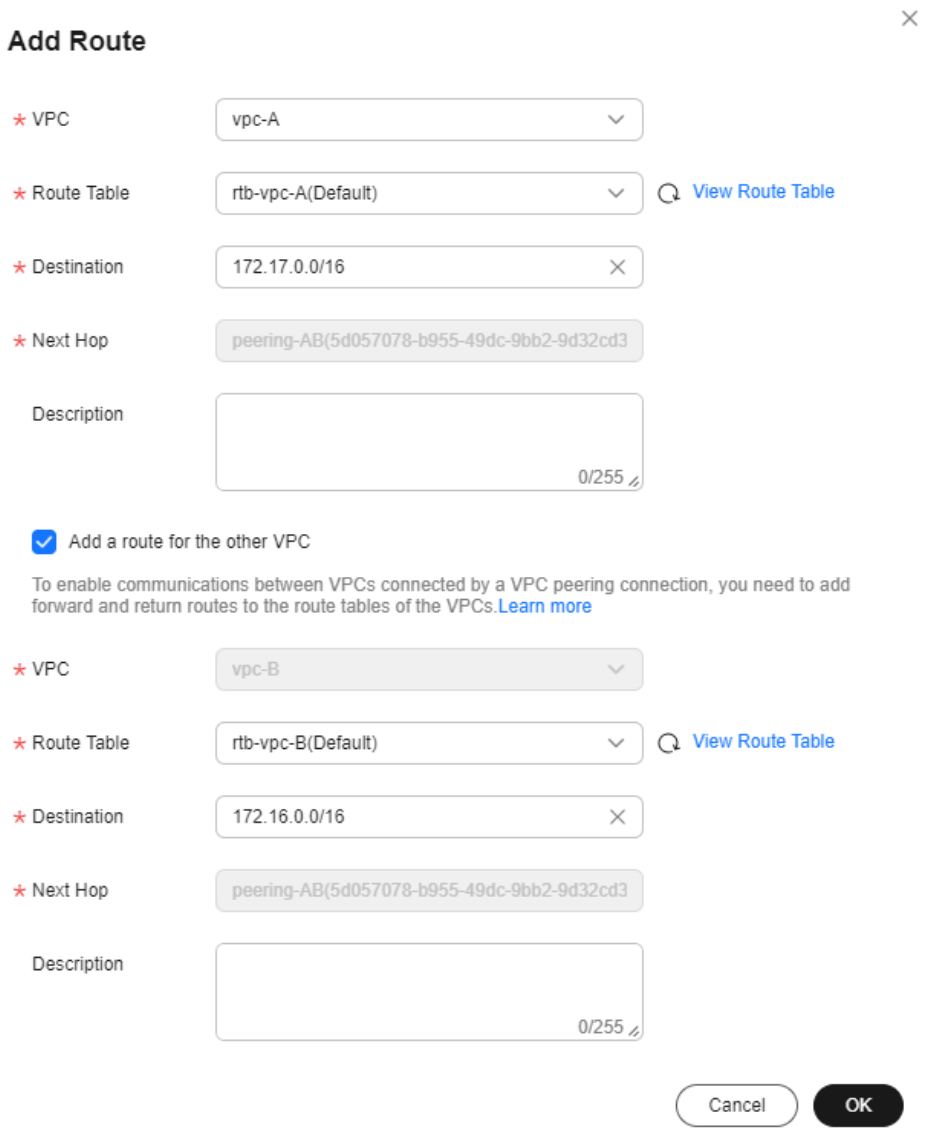
5. In the displayed dialog box, click **Add Now**. On the displayed page about the VPC peering connection details, go to **Step 2: Add Routes for the VPC Peering Connection** to add a route.

Step 2: Add Routes for the VPC Peering Connection

1. In the lower part of the VPC peering connection details page, click **Add Route**.

   The **Add Route** dialog box is displayed.

Figure 8-28 Adding a route



2. Add routes to the route tables as prompted.
   Table 8-59 describes the parameters.

Table 8-59 Parameter description

Parameter	Description	Example Value
VPC	Select a VPC that is connected by the VPC peering connection.	vpc-A

Parameter	Description	Example Value
Route Table	Select the route table of the VPC. The route will be added to this route table. Each VPC comes with a default route table to control the outbound traffic from the subnets in the VPC. In addition to the default route table, you can also create a custom route table and associate it with the subnets in the VPC. Then, the custom route table controls outbound traffic of the subnets. • If there is only the default route table in the drop-down list, select the default route table. • If there are both default and custom route tables in drop-down list, select the route table associated with the subnet connected by the VPC peering connection.	rtb-vpc-A (Default)
Destination	An IP address or address range in the VPC being connected by the VPC peering connection. The value can be a VPC CIDR block, subnet CIDR block, or ECS IP address. For details about the route configuration example, see **VPC Peering Connection Usage Examples**.	vpc-B CIDR block: 172.17.0.0/16
Next Hop	The default value is the current VPC peering connection. You do not need to specify this parameter.	peering-AB
Description	Supplementary information about the route. This parameter is optional. The description can contain a maximum of 255 characters and cannot contain angle brackets (< or >).	Route from vpc-A to vpc-B
Add a route for the other VPC	If you select this option, you can also add a route for the other VPC connected by the VPC peering connection. To enable communications between VPCs connected by a VPC peering connection, you need to add both forward and return routes to the route tables of the VPCs. For details, see **VPC Peering Connection Usage Examples**.	Selected

Parameter	Description	Example Value
VPC	By default, the system selects the VPC connected by the VPC peering connection. You do not need to specify this parameter.	vpc-B
Route Table	Select the route table of the VPC. The route will be added to this route table. Each VPC comes with a default route table to control the outbound traffic from the subnets in the VPC. In addition to the default route table, you can also create a custom route table and associate it with the subnets in the VPC. Then, the custom route table controls outbound traffic of the subnets. • If there is only the default route table in the drop-down list, select the default route table. • If there are both default and custom route tables in drop-down list, select the route table associated with the subnet connected by the VPC peering connection.	rtb-vpc-B (Default)
Destination	An IP address or address range in the other VPC connected by the VPC peering connection. The value can be a VPC CIDR block, subnet CIDR block, or ECS IP address. For details about the route configuration example, see **VPC Peering Connection Usage Examples**.	vpc-A CIDR block: 172.16.0.0/16
Next Hop	The default value is the current VPC peering connection. You do not need to specify this parameter.	peering-AB
Description	Supplementary information about the route. This parameter is optional. The description can contain a maximum of 255 characters and cannot contain angle brackets (< or >).	Route from vpc-B to vpc-A

3. Click **OK**.

   You can view the routes in the route list.

Step 3: Configure Security Group Rules for Instances in Local and Peer VPCs

When configuring security group rules to control traffic in and out of **ECS-A01** and **RDS-B01**, you can set **Template** to **Custom**. Table 8-60 shows the preset security group rules.

Table 8-60 Preset security group rules (Custom rules)

Direction	Action	Type	Protocol & Port	Source/ Destination	Description
Inbound	Allow	IPv4	All	Source: current security group	Allows the instances in the security group to communicate with each other over any IPv4 protocol and port.
Inbound	Allow	IPv6	All	Source: current security group	Allows the instances in the security group to communicate with each other over any IPv6 protocol and port.
Outbound	Allow	IPv4	All	Destination: 0.0.0.0/0	Allows all IPv4 traffic from the instances in the security group to external resources over any protocol and port.
Outbound	Allow	IPv6	All	Destination: ::/0	Allows all IPv6 traffic from the instances in the security group to external resources over any protocol and port.

As you can see, the above preset rules only allow instances in the security group to communicate with each other. To allow external traffic to access the instances in the security group, add inbound rules by referring to **Adding a Security Group Rule**.

- If the instances in the local and peer VPCs are associated with the same security group, they can communicate with each other.

  For example, if **ECS-A01** and **RDS-B01** are associated with security group **Sg-AB**, you only need to perform the operations in **1** to allow remote logins.
- If the instances in the local and peer VPCs are associated with different security groups, they cannot communicate with each other unless you add rules to allow them to.

  For example, if **ECS-A01** is associated with security group **Sg-A**, and **RDS-B01** is associated with security group **Sg-B**, you need to perform the operations in **1** and **2** to allow remote logins.

1. Add security group rules in **Table 8-61** to allow remote logins.

Table 8-61 Security group rules for remote logins

Direction	Action	Type	Protocol & Port	Source	Description
Inbound	Allow	IPv4	TCP: 22	IP address: 0.0.0.0/0	Allows any IPv4 address to remotely log in to the Linux instances in **Sg-AB** over SSH port 22.
Inbound	Allow	IPv4	TCP: 3389	IP address: 0.0.0.0/0	Allows any IPv4 address to remotely log in to the Windows instances in **Sg-AB** over RDP port 3389.

NOTICE

If the source of an inbound rule is set to 0.0.0.0/0, all external IP addresses are allowed to remotely log in to your cloud server. Exposing port 22 or 3389 to the public network will leave your instances vulnerable to network risks. To address this issue, set the source to a trusted IP address, for example, the IP address of your local PC.

2. (Optional) Add security rules to allow the instances in **Sg-A** and **Sg-B** to communicate with each other.

   Select either of the following solution based on your service requirements.

   – Solution 1 in **Table 8-62**: Set **Source** to the CIDR block of peer VPC or subnet.

   Table 8-62 Security group rules (CIDR block as the source)

Security Group	Direction	Action	Type	Protocol & Port	Source	Description
Sg-A	Inbound	Allow	IPv4	All	IP address: 172.17.0.0/16 (**vpc-B** CIDR block)	Allows traffic from 172.17.0.0/16 to access instances in **Sg-A** over any IPv4 protocol and port.
Sg-B	Inbound	Allow	IPv4	All	IP address: 172.16.0.0/16 (**vpc-A** CIDR block)	Allows traffic from 172.16.0.0/16 to access instances in **Sg-B** over any IPv4 protocol and port.

– Solution 2 in **Table 8-63**: Set **Source** to **Sg-A** and **Sg-B**.

Table 8-63 Security group rules (security group as the source)

Security Group	Direction	Action	Type	Protocol & Port	Source	Description
Sg-A	Inbound	Allow	IPv4	All	Sg-B	Allows instances in **Sg-B** to access those in **Sg-A** over any IPv4 protocol and port.
Sg-B	Inbound	Allow	IPv4	All	Sg-A	Allows instances in **Sg-A** to access those in **Sg-B** over any IPv4 protocol and port.

Step 4: Verify Network Connectivity

After you add routes for the VPC peering connection, verify communications between the local and peer VPCs.

1. Log in to **ECS-A01** in the local VPC.

   Multiple methods are available for logging in to an ECS. For details, see **Logging In to an ECS**.

2. Check whether **ECS-A01** can communicate with **RDS-B01**.

   ping *<peer-server-IP-address>*

   Example command:

   ping 172.17.0.21

   If information similar to the following is displayed, **ECS-A01** and **RDS-B01** can communicate with each other, and the VPC peering connection between **VPC-A** and **VPC-B** is successfully created.

```
[root@ecs-A01 ~]# ping 172.17.0.21
PING 172.17.0.21 (172.17.0.21) 56(84) bytes of data.
64 bytes from 172.17.0.21: icmp_seq=1 ttl=64 time=0.849 ms
64 bytes from 172.17.0.21: icmp_seq=2 ttl=64 time=0.455 ms
64 bytes from 172.17.0.21: icmp_seq=3 ttl=64 time=0.385 ms
64 bytes from 172.17.0.21: icmp_seq=4 ttl=64 time=0.372 ms
...
--- 172.17.0.21 ping statistics ---
```

NOTICE

In this example, ECS-A01 and RDS-B01 are in the same security group. If the instances in different security groups, you need to add inbound rules to allow access from the peer security group. For details, see **Enabling Communications Between Instances in Different Security Groups**.

If VPCs connected by a VPC peering connection cannot communicate with each other, refer to **Why Did Communication Fail Between VPCs That Were Connected by a VPC Peering Connection?**

8.4 Creating a VPC Peering Connection to Connect Two VPCs in Different Accounts

Scenarios

Two VPCs from the same region cannot communicate with each other by default, but you can use a VPC peering connection to connect them.

The following describes how to create a VPC peering connection to connect two VPCs, **vpc-A** in one account and **vpc-B** in another account. In this way, instances (the service server **ECS-A01** and database server **RDS-B01** in this example) in the two VPCs can communicate with each other.

The procedure is as follows:

Step 1: Create a VPC Peering Connection

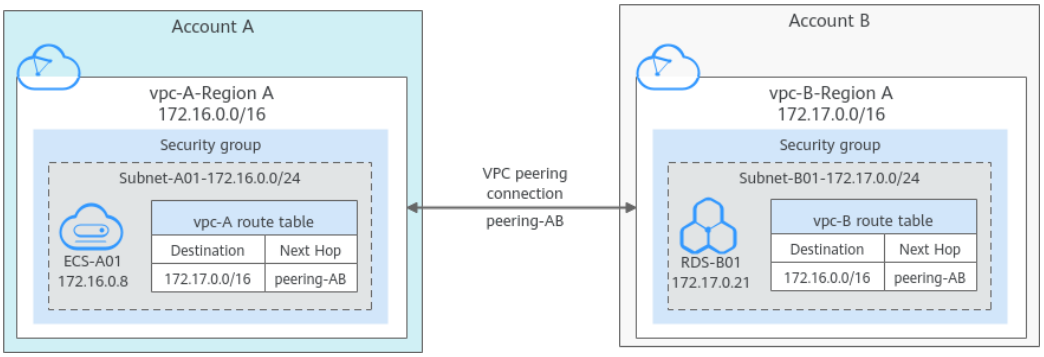
Step 2: Peer Account Accepts the VPC Peering Connection Request

Step 3: Add Routes for the VPC Peering Connection

Step 4: Configure Security Group Rules for Instances in Local and Peer VPCs

Step 5: Verify Network Connectivity

Figure 8-29 Connecting two VPCs in different accounts using a VPC peering connection



NOTICE

Currently, VPC peering connections are free.

Notes and Constraints

- Only one VPC peering connection can be created between two VPCs at the same time.
- A VPC peering connection can only connect VPCs in the same region.
  - A VPC peering connection can enable a VPC created on the Huawei Cloud Chinese Mainland website to connect to one created on the Huawei

Cloud International website, but the VPCs must be in the same region. For example, if the VPC created on the Chinese Mainland website is in the CN-Hong Kong region, then the VPC created on the International website must also be in the CN-Hong Kong region.

  - If you want to connect VPCs in different regions, you can use **Cloud Connect**.
  - If you only need a few ECSs in different regions to communicate with each other, you can **assign and bind EIPs to the ECSs**.
- If the local and peer VPCs have overlapping CIDR blocks, the VPC peering connection may not be usable.

  In this case, you can configure the network by referring to **VPC Peering Connection Usage Examples**.
- For a VPC peering connection between VPCs in different accounts:
  - If account A initiates a request to create a VPC peering connection with a VPC in account B, the VPC peering connection takes effect only after account B accepts the request.
  - To ensure network security, do not accept VPC peering connections from untrusted accounts.

Prerequisites

You have two VPCs in the same region, but they are from different accounts. If you want to create one, see **Creating a VPC and Subnet**.

Step 1: Create a VPC Peering Connection

1. Go to the **VPC peering connection list page**.
2. In the upper right corner of the page, click **Create VPC Peering Connection**.

   The **Create VPC Peering Connection** page is displayed.
3. Configure the parameters as prompted.

   For details, see **Table 8-64**.

Figure 8-30 Creating a VPC peering connection

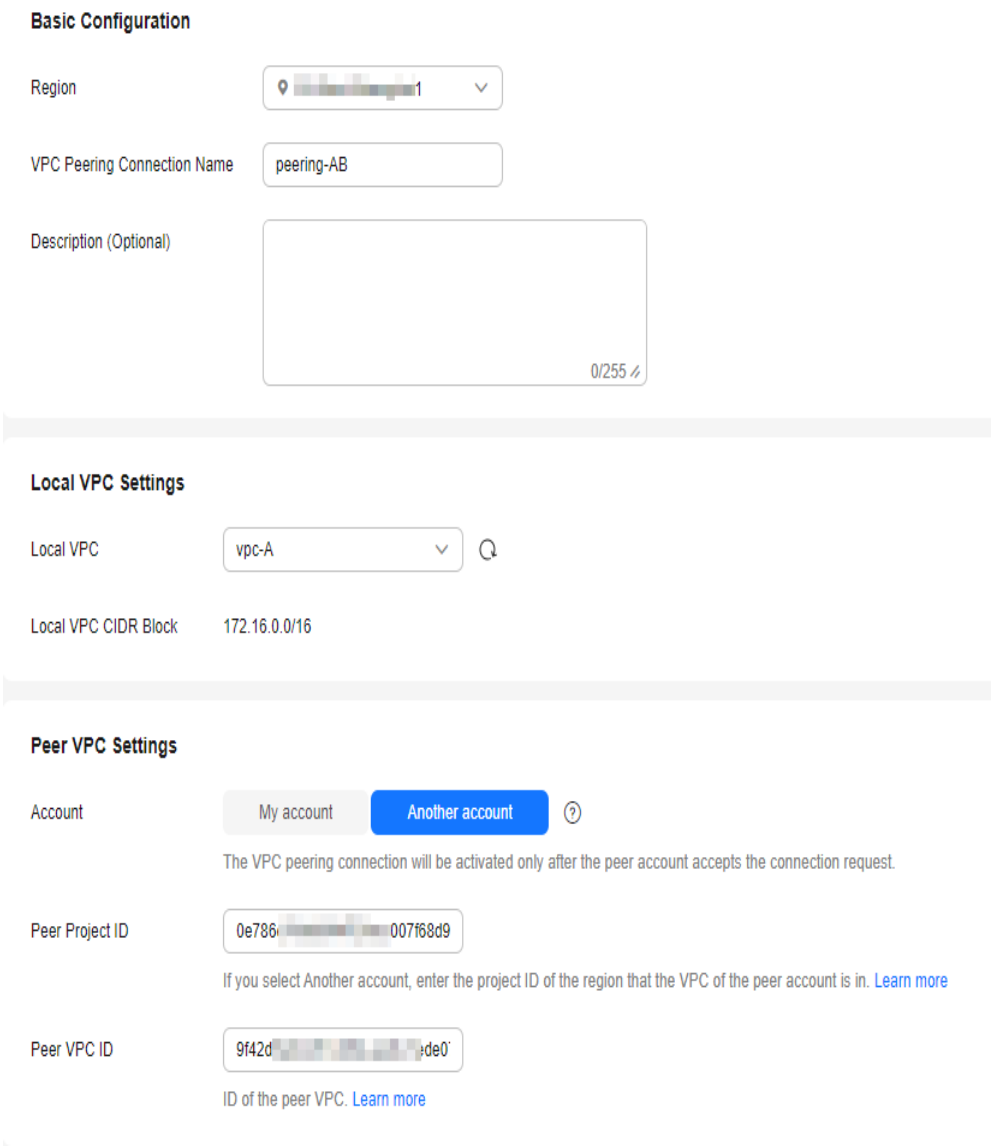


Table 8-64 Parameters for creating a VPC peering connection

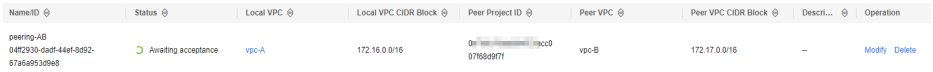
Parameter	Description	Example Value
Region	Mandatory The region where the VPC peering connection is created. Select the region nearest to you to ensure the lowest latency possible.	CN-Hong Kong

Parameter	Description	Example Value
VPC Peering Connection Name	Mandatory Enter a name for the VPC peering connection. The name can contain a maximum of 64 characters, including letters, digits, hyphens (-), and underscores (_).	peering-AB
Description (Optional)	Optional Enter a description of the VPC peering connection in the text box as required. The description can contain a maximum of 255 characters and cannot contain angle brackets (< or >).	peering-AB connects **vpc-A** and **vpc-B**.
Local VPC	Mandatory VPC at one end of the VPC peering connection. You can select one from the drop-down list.	vpc-A
Local VPC CIDR Block	CIDR block of the selected local VPC	172.16.0.0/16
Account	Mandatory • Options: **My account** and **Another account** • Select **Another account**.	Another account
Peer Project ID	This parameter is mandatory if **Account** is set to **Another account**. The project ID of the region that the peer VPC resides. For details about how to obtain the project ID, see **Obtaining the Peer Project ID of a VPC Peering Connection**.	Project ID of **vpc-B** in region A: 067cf8aecf3XXX08322f13b
Peer VPC ID	This parameter is mandatory if **Account** is set to **Another account**. ID of the VPC at the other end of the VPC peering connection. For details about how to obtain the ID, see **Obtaining a VPC ID**.	**vpc-B** ID: 17cd7278-XXX-530c952dcf35

4. Click **Create Now**.

- If the message "Invalid VPC ID and project ID." is displayed, check whether the project ID and VPC ID are correct.
  - Peer Project ID: The value must be the project ID of the region where the peer VPC resides.
  - The local and peer VPCs must be in the same region.
- If the status of the created VPC peering connection is **Awaiting acceptance**, go to **Step 2: Peer Account Accepts the VPC Peering Connection Request**.

Figure 8-31 Awaiting acceptance



Step 2: Peer Account Accepts the VPC Peering Connection Request

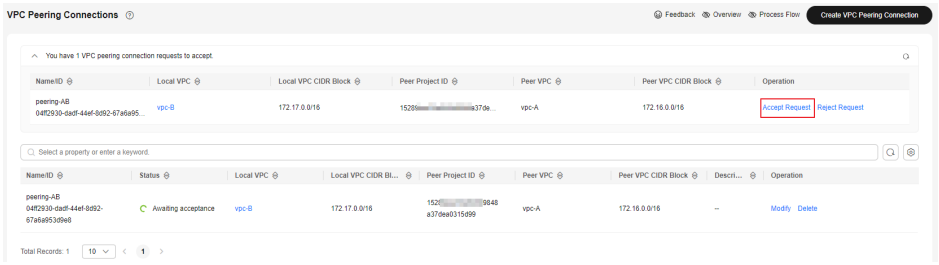
After you create a VPC peering connection with a VPC in another account, you need to contact the peer account to accept the VPC peering connection request. In this example, account A notifies account B to accept the request. Account B needs to:

1. Log in to the management console.
2. Click ☰ in the upper left corner and choose **Networking** > **Virtual Private Cloud**.

   The **Virtual Private Cloud** page is displayed.
3. In the navigation pane on the left, choose **Virtual Private Cloud** > **VPC Peering Connections**.

   The VPC peering connection list is displayed.
4. In the upper part of the VPC peering connection list, locate the VPC peering connection request to be accepted.

Figure 8-32 Accept Request



5. Locate the row that contains the target VPC peering connection and click **Accept Request** in the **Operation** column.

   After the status of the VPC peering connection changes to **Accepted**, the VPC peering connection is created.
6. Go to **Step 3: Add Routes for the VPC Peering Connection**.

Step 3: Add Routes for the VPC Peering Connection

To enable communications between VPCs connected by a VPC peering connection, you need to add both forward and return routes to the route tables of the VPCs. For details, see **VPC Peering Connection Usage Examples**.

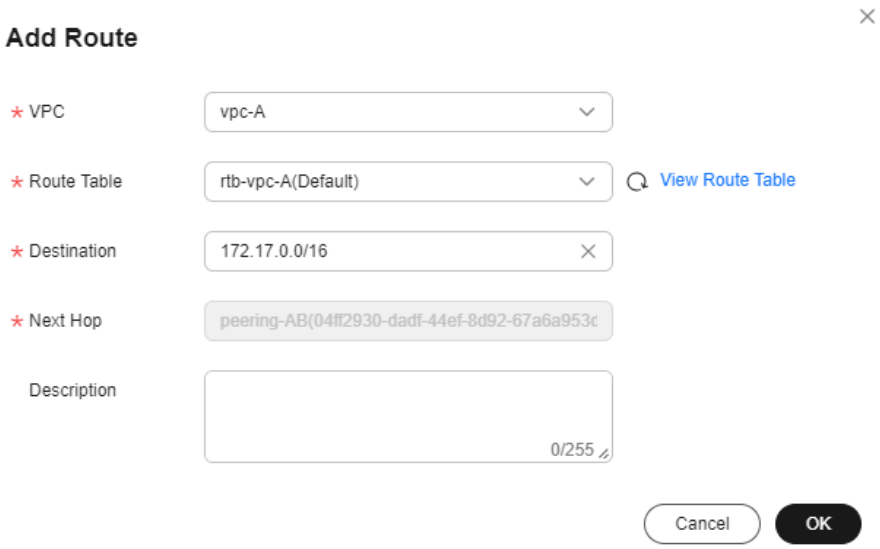
Both accounts need to add a route to the route table of their VPC. In this example, account A adds a route to the route table of VPC-A, and account B adds a route to the route table of VPC-B.

1. Add routes to the route table of the local VPC:
   a. In the VPC peering connection list of the local account, click the name of the target VPC peering connection.

      The page showing the VPC peering connection details is displayed.
   b. In the lower part of the VPC peering connection details page, click **Add Route**.

      The **Add Route** dialog box is displayed.

      Figure 8-33 Add Route

   c. Add routes to the route tables as prompted.

      Table 8-65 describes the parameters.

      Table 8-65 Parameter description

Parameter	Description	Example Value
VPC	By default, the VPC in the current account is selected. You do not need to select a VPC.	vpc-A

Parameter	Description	Example Value
Route Table	Select the route table of the VPC. The route will be added to this route table. Each VPC comes with a default route table to control the outbound traffic from the subnets in the VPC. In addition to the default route table, you can also create a custom route table and associate it with the subnets in the VPC. Then, the custom route table controls outbound traffic of the subnets. • If there is only the default route table in the drop-down list, select the default route table. • If there are both default and custom route tables in drop-down list, select the route table associated with the subnet connected by the VPC peering connection.	rtb-vpc-A (Default route table)
Destination	An IP address or address range in the VPC being connected by the VPC peering connection. The value can be a VPC CIDR block, subnet CIDR block, or ECS IP address. For details about the route configuration example, see **VPC Peering Connection Usage Examples**.	vpc-B CIDR block: 172.17.0.0/16
Next Hop	The default value is the current VPC peering connection. You do not need to specify this parameter.	peering-AB
Description	Supplementary information about the route. This parameter is optional. The description can contain a maximum of 255 characters and cannot contain angle brackets (< or >).	Route from vpc-A to vpc-B

d. Click **OK**.

You can view the routes in the route list.

2. Add routes to the route table of the peer VPC:

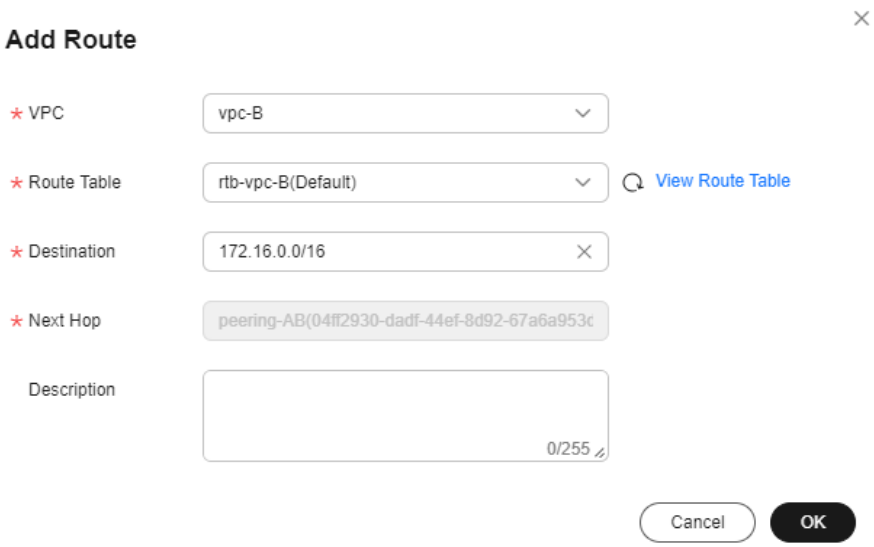
a. In the VPC peering connection list of the peer account, click the name of the target VPC peering connection.

The page showing the VPC peering connection details is displayed.

b. In the lower part of the VPC peering connection details page, click **Add Route**.

The **Add Route** dialog box is displayed.

Figure 8-34 Add Route



c. Add routes to the route table as prompted.

Table 8-66 describes the parameters.

Table 8-66 Parameter description

Parameter	Description	Example Value
VPC	By default, the VPC in the current account is selected. You do not need to select a VPC.	vpc-B

Parameter	Description	Example Value
Route Table	Select the route table of the VPC. The route will be added to this route table. Each VPC comes with a default route table to control the outbound traffic from the subnets in the VPC. In addition to the default route table, you can also create a custom route table and associate it with the subnets in the VPC. Then, the custom route table controls outbound traffic of the subnets. • If there is only the default route table in the drop-down list, select the default route table. • If there are both default and custom route tables in drop-down list, select the route table associated with the subnet connected by the VPC peering connection.	rtb-vpc-B (Default route table)
Destination	An IP address or address range in the VPC being connected by the VPC peering connection. The value can be a VPC CIDR block, subnet CIDR block, or ECS IP address. For details about the route configuration example, see **VPC Peering Connection Usage Examples**.	vpc-A CIDR block: 172.16.0.0/16
Next Hop	The default value is the current VPC peering connection. You do not need to specify this parameter.	peering-AB
Description	Supplementary information about the route. This parameter is optional. The description can contain a maximum of 255 characters and cannot contain angle brackets (< or >).	Route from vpc-B to vpc-A

d. Click **OK**.

You can view the routes in the route list.

Step 4: Configure Security Group Rules for Instances in Local and Peer VPCs

When configuring security group rules to control traffic in and out of **ECS-A01** and **RDS-B01**, you can set **Template** to **Custom**. Table 8-67 shows the preset security group rules.

Table 8-67 Preset security group rules (Custom rules)

Direction	Action	Type	Protocol & Port	Source/ Destination	Description
Inbound	Allow	IPv4	All	Source: current security group	Allows the instances in the security group to communicate with each other over any IPv4 protocol and port.
Inbound	Allow	IPv6	All	Source: current security group	Allows the instances in the security group to communicate with each other over any IPv6 protocol and port.
Outbound	Allow	IPv4	All	Destination: 0.0.0.0/0	Allows all IPv4 traffic from the instances in the security group to external resources over any protocol and port.
Outbound	Allow	IPv6	All	Destination: ::/0	Allows all IPv6 traffic from the instances in the security group to external resources over any protocol and port.

As you can see, the above preset rules only allow instances in the security group to communicate with each other. To allow external traffic to access the instances in the security group, add inbound rules by referring to **Adding a Security Group Rule**.

The instances in the local and peer VPCs in different accounts are associated with different security groups, so they cannot communicate with each other unless you add rules to allow them to. For example, if **ECS-A01** is associated with security group **Sg-A**, and **RDS-B01** is associated with security group **Sg-B**, you need to perform the following operations to allow remote logins:

1. Add security group rules in **Table 8-68** to allow remote logins.

Table 8-68 Security group rules for remote logins

Direction	Action	Type	Protocol & Port	Source	Description
Inbound	Allow	IPv4	TCP: 22	IP address: 0.0.0.0/0	Allows any IPv4 address to remotely log in to the Linux instances in **Sg-AB** over SSH port 22.

Direction	Action	Type	Protocol & Port	Source	Description
Inbound	Allow	IPv4	TCP: 3389	IP address: 0.0.0.0/0	Allows any IPv4 address to remotely log in to the Windows instances in **Sg-AB** over RDP port 3389.

NOTICE

If the source of an inbound rule is set to 0.0.0.0/0, all external IP addresses are allowed to remotely log in to your cloud server. Exposing port 22 or 3389 to the public network will leave your instances vulnerable to network risks. To address this issue, set the source to a trusted IP address, for example, the IP address of your local PC.

2. Add security group rules in **Table 8-69** to enable instances in **Sg-A** and **Sg-B** to communicate with each other.

   Set **Source** to the CIDR block of peer VPC or subnet.

Table 8-69 Security group rules (CIDR block as the source)

Security Group	Direction	Action	Type	Protocol & Port	Source	Description
Sg-A	Inbound	Allow	IPv4	All	IP address: 172.17.0.0/16 (**vpc-B** CIDR block)	Allows traffic from 172.17.0.0/16 to access instances in **Sg-A** over any IPv4 protocol and port.
Sg-B	Inbound	Allow	IPv4	All	IP address: 172.16.0.0/16 (**vpc-A** CIDR block)	Allows traffic from 172.16.0.0/16 to access instances in **Sg-B** over any IPv4 protocol and port.

Step 5: Verify Network Connectivity

After you add routes for the VPC peering connection, verify communications between the local and peer VPCs.

1. Log in to **ECS-A01** in the local VPC.

   Multiple methods are available for logging in to an ECS. For details, see **Logging In to an ECS**.

2. Check whether **ECS-A01** can communicate with **RDS-B01**.

   ping *<peer-server-IP-address>*

   Example command:

   ping 172.17.0.21

   If information similar to the following is displayed, **ECS-A01** and **RDS-B01** can communicate with each other, and the VPC peering connection between **VPC-A** and **VPC-B** is successfully created.

```
[root@ecs-A01 ~]# ping 172.17.0.21
PING 172.17.0.21 (172.17.0.21) 56(84) bytes of data.
64 bytes from 172.17.0.21: icmp_seq=1 ttl=64 time=0.849 ms
64 bytes from 172.17.0.21: icmp_seq=2 ttl=64 time=0.455 ms
64 bytes from 172.17.0.21: icmp_seq=3 ttl=64 time=0.385 ms
64 bytes from 172.17.0.21: icmp_seq=4 ttl=64 time=0.372 ms
...
--- 172.17.0.21 ping statistics ---
```

NOTICE

In this example, ECS-A01 and RDS-B01 are in the same security group. If the instances in different security groups, you need to add inbound rules to allow access from the peer security group. For details, see **Enabling Communications Between Instances in Different Security Groups**.

If VPCs connected by a VPC peering connection cannot communicate with each other, refer to **Why Did Communication Fail Between VPCs That Were Connected by a VPC Peering Connection?**

8.5 Obtaining the Peer Project ID of a VPC Peering Connection

Scenarios

If you create a VPC peering connection between two VPCs in different accounts, you can refer to this section to obtain the project ID of the region that the peer VPC resides.

Procedure

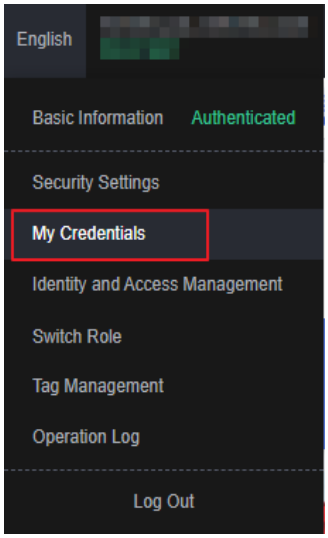
1. Log in to the management console.

   The owner of the peer account logs in to the management console.

2. In the upper right corner of the page, select **My Credentials** from the username drop-down list.

   The **My Credentials** page is displayed.

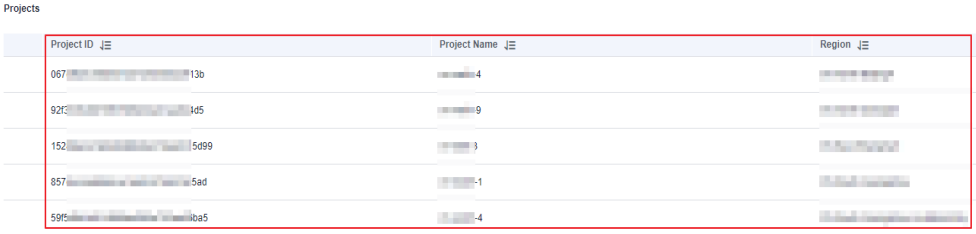
Figure 8-35 My Credentials



3. In the project list, obtain the project ID.

   Locate the region of the peer VPC and obtain the project ID corresponding to the region.

Figure 8-36 Project ID



8.6 Modifying a VPC Peering Connection

Scenarios

This section describes how to modify the basic information about a VPC peering connection, including its name and description.

Either owner of a VPC in a peering connection can modify the VPC peering connection in any state.

Procedure

1. Log in to the management console.
2. Click in the upper left corner and select the desired region and project.
3. Click in the upper left corner and choose **Networking** > **Virtual Private Cloud**.

   The **Virtual Private Cloud** page is displayed.

4. In the navigation pane on the left, choose **Virtual Private Cloud** > **VPC Peering Connections**.

   The VPC peering connection list is displayed.

5. In the VPC peering connection list, locate the row that contains the target VPC peering connection and click **Modify** in the **Operation** column.

   The **Modify VPC Peering Connection** dialog box is displayed.

6. Modify the VPC peering connection information and click **OK**.

8.7 Viewing VPC Peering Connections

Scenarios

This section describes how to view basic information about a VPC peering connection, including the connection name, status, and information about the local and peer VPCs.

If a VPC peering connection is created between two VPCs in different accounts, both the local and peer accounts can view information about the VPC peering connection.

Procedure

1. Log in to the management console.
2. Click  in the upper left corner and select the desired region and project.
3. Click  in the upper left corner and choose **Networking** > **Virtual Private Cloud**.

   The **Virtual Private Cloud** page is displayed.

4. In the navigation pane on the left, choose **Virtual Private Cloud** > **VPC Peering Connections**.

   The VPC peering connection list is displayed.

5. In the VPC peering connection list, click the name of the target VPC peering connection.

   On the displayed page, view details about the VPC peering connection.

8.8 Deleting a VPC Peering Connection

Scenarios

This section describes how to delete a VPC peering connection.

Either owner of a VPC in a peering connection can delete the VPC peering connection in any state.

Notes and Constraints

The owner of either VPC in a peering connection can delete the VPC peering connection at any time. Deleting a VPC peering connection will also delete all

information about this connection, including the routes in the local and peer VPC route tables added for the connection.

Procedure

1. Log in to the management console.
2. Click in the upper left corner and select the desired region and project.
3. Click in the upper left corner and choose **Networking** > **Virtual Private Cloud**.

   The **Virtual Private Cloud** page is displayed.
4. In the navigation pane on the left, choose **Virtual Private Cloud** > **VPC Peering Connections**.

   The VPC peering connection list is displayed.
5. In the VPC peering connection list, locate the row that contains the target VPC peering connection and click **Delete** in the **Operation** column.

   A confirmation dialog box is displayed.
6. Click **OK**.

8.9 Modifying Routes Configured for a VPC Peering Connection

Scenarios

This section describes how to modify the routes added for a VPC peering connection in the route tables of the local and peer VPCs.

- **Modifying Routes of a VPC Peering Connection Between VPCs in the Same Account**
- **Modifying Routes of a VPC Peering Connection Between VPCs in Different Accounts**

You can follow the instructions provided in this section to modify routes based on your requirements.

Modifying Routes of a VPC Peering Connection Between VPCs in the Same Account

1. Log in to the management console.
2. Click in the upper left corner and select the desired region and project.
3. Click in the upper left corner and choose **Networking** > **Virtual Private Cloud**.

   The **Virtual Private Cloud** page is displayed.
4. In the navigation pane on the left, choose **Virtual Private Cloud** > **VPC Peering Connections**.

   The VPC peering connection list is displayed.

5. In the VPC peering connection list, click the name of the target VPC peering connection.

   The page showing the VPC peering connection details is displayed.
6. In the route list, click the route table hyperlink of the route.

   The route table details page is displayed.
7. In the route list, locate the route and click **Modify** in the **Operation** column.
8. Modify the route and click **OK**.

Modifying Routes of a VPC Peering Connection Between VPCs in Different Accounts

Only the account owner of a VPC can modify the routes added for the connection.

1. Log in to the management console using the account of the local VPC and modify the route of the local VPC:
   a. Click  in the upper left corner and select the desired region and project.
   b. Click  in the upper left corner and choose **Networking** > **Virtual Private Cloud**.

      The **Virtual Private Cloud** page is displayed.
   c. In the navigation pane on the left, choose **Virtual Private Cloud** > **VPC Peering Connections**.

      The VPC peering connection list is displayed.
   d. In the VPC peering connection list, click the name of the target VPC peering connection.

      The page showing the VPC peering connection details is displayed.
   e. In the route list, click the name of the target route table in the **Route Table** column.

      The route table details page is displayed.
   f. In the route list, locate the route and click **Modify** in the **Operation** column.
   g. Modify the route and click **OK**.
2. Log in to the management console using the account of the peer VPC and modify the route of the peer VPC by referring to **1**.

8.10 Viewing Routes Configured for a VPC Peering Connection

Scenarios

This section describes how to view the routes added to the route tables of local and peer VPCs of a VPC peering connection.

- **Viewing Routes of a VPC Peering Connection Between VPCs in the Same Account**

- **Viewing Routes of a VPC Peering Connection Between VPCs in Different Accounts**

If two VPCs cannot communicate through a VPC peering connection, you can check the routes added for the local and peer VPCs by following the instructions provided in this section.

Viewing Routes of a VPC Peering Connection Between VPCs in the Same Account

1. Log in to the management console.
2. Click  in the upper left corner and select the desired region and project.
3. Click  in the upper left corner and choose **Networking** > **Virtual Private Cloud**.

   The **Virtual Private Cloud** page is displayed.
4. In the navigation pane on the left, choose **Virtual Private Cloud** > **VPC Peering Connections**.

   The VPC peering connection list is displayed.
5. In the VPC peering connection list, click the name of the target VPC peering connection.

   The page showing the VPC peering connection details is displayed.
6. In the route list, view the route information.

   You can view the route destination, VPC, next hop, route table, and more.

Viewing Routes of a VPC Peering Connection Between VPCs in Different Accounts

Only the account owner of a VPC in a VPC peering connection can view the routes added for the connection.

1. Log in to the management console using the account of the local VPC and view the route of the local VPC:
   a. Click  in the upper left corner and select the desired region and project.
   b. Click  in the upper left corner and choose **Networking** > **Virtual Private Cloud**.

      The **Virtual Private Cloud** page is displayed.
   c. In the navigation pane on the left, choose **Virtual Private Cloud** > **VPC Peering Connections**.

      The VPC peering connection list is displayed.
   d. In the VPC peering connection list, click the name of the target VPC peering connection.

      The page showing the VPC peering connection details is displayed.
   e. In the route list, view the route information.

      You can view the route destination, VPC, next hop, route table, and more.

2. Log in to the management console using the account of the peer VPC and view the route of the peer VPC by referring to **1**.

8.11 Deleting Routes Configured for a VPC Peering Connection

Scenarios

This section describes how to delete routes from the route tables of the local and peer VPCs connected by a VPC peering connection.

- **Deleting Routes of a VPC Peering Connection Between VPCs in the Same Account**
- **Deleting Routes of a VPC Peering Connection Between VPCs in Different Accounts**

Deleting Routes of a VPC Peering Connection Between VPCs in the Same Account

1. Log in to the management console.
2. Click  in the upper left corner and select the desired region and project.
3. Click  in the upper left corner and choose **Networking** > **Virtual Private Cloud**.

   The **Virtual Private Cloud** page is displayed.
4. In the navigation pane on the left, choose **Virtual Private Cloud** > **VPC Peering Connections**.

   The VPC peering connection list is displayed.
5. In the VPC peering connection list, click the name of the target VPC peering connection.

   The page showing the VPC peering connection details is displayed.
6. In the route list, locate the route and click **Delete** in the **Operation** column.

   A confirmation dialog box is displayed.
7. Confirm the information and click **OK**.

Deleting Routes of a VPC Peering Connection Between VPCs in Different Accounts

Only the account owner of a VPC in a VPC peering connection can delete the routes added for the connection.

1. Log in to the management console using the account of the local VPC and delete the route of the local VPC:

   a. Click  in the upper left corner and select the desired region and project.

   b. Click  in the upper left corner and choose **Networking** > **Virtual Private Cloud**.

The **Virtual Private Cloud** page is displayed.

c. In the navigation pane on the left, choose **Virtual Private Cloud** > **VPC Peering Connections**.

The VPC peering connection list is displayed.

d. In the VPC peering connection list, click the name of the target VPC peering connection.

The page showing the VPC peering connection details is displayed.

e. In the route list, locate the route and click **Delete** in the **Operation** column.

A confirmation dialog box is displayed.

f. Confirm the information and click **OK**.

2. Log in to the management console using the account of the peer VPC and delete the route of the peer VPC by referring to **1**.

9 VPC Sharing

9.1 VPC Sharing Overview

What Is VPC Sharing?

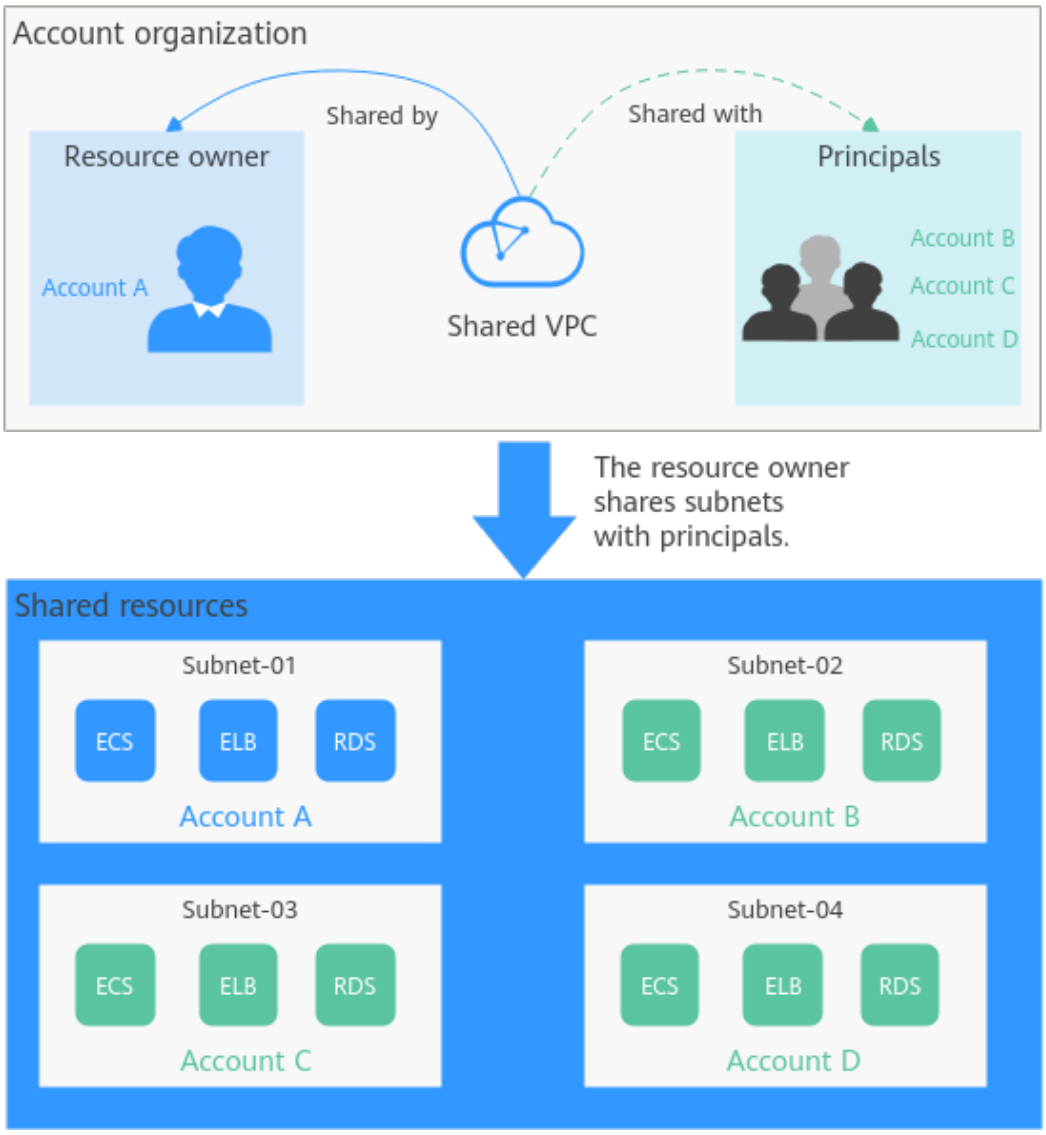
VPC sharing allows multiple accounts to create and manage cloud resources, such as ECSs, load balancers, and RDS instances, in one VPC. With Resource Access Manager (RAM), you can share subnets in a VPC with one or more accounts so you can centrally manage resources in multiple accounts. This helps you improve resource management efficiency and reduce O&M costs.

The following describes how you can share subnets among several accounts, as shown in **Figure 9-1**.

- Account A: IT management account of the enterprise and the owner of the VPC and subnets.

  Account A creates a VPC and four subnets and shares these subnets with other accounts. Account A creates resources in Subnet-01.
- Account B: service account of the enterprise and the principal of the shared subnet. Account B creates resources in Subnet-02.
- Account C: service account of the enterprise and the principal of the shared subnet. Account C creates resources in Subnet-03.
- Account D: service account of the enterprise and the principal of the shared subnet. Account D creates resources in Subnet-04.

Figure 9-1 Application scenario



NOTICE

The subnets of the owner and those of the principals are in the same VPC, so resources in these subnets can communicate with each other by default. However, if the resources in the subnets are associated with different security groups, the resources are isolated from each other. If you want the resources to communicate with each other, you need to add security group rules by referring to **Adding a Security Group Rule**.

For example, to allow ECSs in accounts A and B to communicate with each other, you need to add inbound rules to their security groups and set the source to the security group in the other account.

Advantages

For basic IT systems of financial enterprises and large enterprises, resources are managed by multiple accounts based on permissions. The following problems may arise from time to time:

- There are multiple accounts, such as network accounts, security accounts, and service accounts. This makes cross-account resource O&M hard and time-consuming.
- The cross-account network configurations result in a complex networking structure, hard user operations, and low efficiency.

To deal with these problems, you can share subnets with multiple accounts. You can organize accounts in an orderly and centralized manner based on organization structure or business model.

- You can create subnets in a VPC under an account and share the subnets with principals. In this way, principals do not need to create VPCs and subnets. Fewer resources and simplified network architecture improves management efficiency and reduces costs.

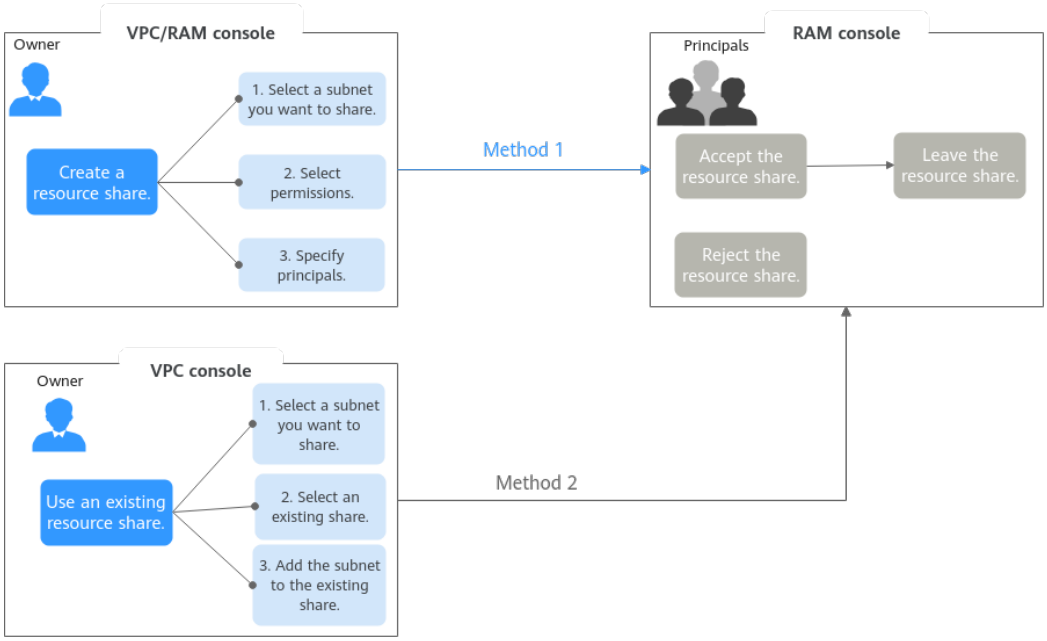
  If there are VPCs in different accounts, VPC peering connections are required for mutual communications among VPCs. With VPC sharing, different accounts can create resources in one VPC. This eliminates the need for configuring VPC peering connections and simplifies the network structure.
- Resources can be centrally managed in one account, which helps enterprises configure service security policies in a centralized manner and better monitor and audit resource usage for higher security.

Process for Sharing a Subnet

Before sharing a subnet, you need to enable the RAM service in your account. For details, see **Resource Access Manager User Guide**.

As the owner of VPC subnets, you can share the subnets with other accounts. Principals need to accept the sharing requests before they use the subnets. **Figure 9-2** shows the process of sharing a subnet.

Figure 9-2 Process for sharing a subnet



You can share a subnet on the RAM or VPC console. For details, see **Table 9-1**.

Table 9-1 The process for sharing a subnet

Method	Description	Reference
Method 1	Creating a resource share to share a subnet 1. The owner locates the subnet to share and clicks its name. On the **Resource Share** tab, the owner can click **Create Share** to go to the RAM console and create a resource share. a. Select the subnet to share. b. Select permissions to grant to principals on the shared subnet. c. Specify one or more principals who can use the shared subnet. 2. On the RAM console, principals accept or reject the resource share. • If principals accept the resource share, they can use the shared subnet. If principals do not want to use the shared subnet, they can leave the resource share. • If principals reject the resource share, they cannot use the subnet.	1. Owner: **Creating a Resource Share** 2. Principals: **Responding to a Resource Sharing Invitation** **Leaving a Resource Share**

Method	Description	Reference
Method B	Using an existing resource share to share a subnet 1. The owner locates the subnet to share and clicks its name. On the **Resource Share** tab, select an existing resource share and add the subnet to it. 2. On the RAM console, principals accept or reject the resource share. • If principals accept the resource share, they can use the shared subnet. If principals do not want to use the shared subnet, they can leave the resource share. • If principals reject the resource share, they cannot use the subnet.	1. Owner: **Sharing a Subnet with Other Accounts** 2. Principals: **Responding to a Resource Sharing Invitation** **Leaving a Resource Share**

Operation Permissions on a Shared Subnet

The owner and principals of a shared subnet have different operation permissions on the subnet and associated resources. For details, see **Table 9-2**.

Table 9-2 Operation permissions on a shared subnet and associated resources

Role	When a Share Is Accepted	When a Share Is Stopped	When the Principals Leave a Share
Owner	• Has operation permissions listed in **Table 9-3**. • Cannot modify or delete resources created by principals, such as ECSs, load balancers, and RDS instances. • Views the information such as the IP address and ID of the resource created by principals on the **IP Addresses** tab of the shared subnet.	• Uses, deletes, and manages all the resources in the VPC. • If principals have resources in the subnet, the owner cannot delete the shared subnet or the VPC where the shared subnet belongs after the share is stopped.	• Uses, deletes, and manages all the resources in the VPC. • If principals have resources in the subnet, the owner cannot delete the shared subnet or the VPC where the shared subnet belongs after the principals leave the share.

Role	When a Share Is Accepted	When a Share Is Stopped	When the Principals Leave a Share
Principal	• Has operation permissions listed in Table 9-3. • Create resources, such as ECSs, load balancers, and RDS instances, in the shared subnets. • Views the information such as the IP address and ID of the resource created by themselves on the **IP Addresses** tab of the shared subnet.	Uses the existing resources created by themselves, but cannot create resources in the shared subnet.	Uses the existing resources created by themselves, but cannot create resources in the shared subnet.

The owner and principals of a shared subnet have different operation permissions on the subnet and associated resources. For details, see Table 9-3.

Table 9-3 Operation permissions on a shared subnet and associated resources (sharing)

Resource	Owner	Principal
VPC	Has all operation permissions on the VPC of a shared subnet.	Only can view the VPC that the shared subnet belongs to, but cannot perform any operations on the VPC.
Subnet	Has all operation permissions on the shared subnet and can view the virtual IP addresses and network interfaces in the shared subnet.	Only can view the shared subnet, but cannot: • Modify the subnet. • Delete the subnet. • Add, modifying, and delete subnet tags. Can assign virtual IP addresses and network interfaces in the subnet.

Resource	Owner	Principal
Route table	Has all operation permissions on the route table.	• Cannot create a route table in the VPC that the shared subnet belongs to. • Can view the route table associated with the shared subnet and the routes in the route table, but cannot perform any operations on the route table or the routes.
Network ACL	Has all operation permissions on the network ACL.	• Can view the network ACL associated with the shared subnet, but cannot perform any operation on the network ACL. • Cannot associate the owner's network ACL with their own subnets.

Resource	Owner	Principal
Security group	• Can create their own security groups. • Only has the operation permissions on their own security groups and cannot perform any operations on the security groups of the principals. • For security groups associated with resources in a shared subnet, the owner can add rules to their own security groups and can set **Source** of the rules to the security groups of the principals. For example, in the shared Subnet-X: – The owner has created ECS-X with security group Sys-X associated. – Principal A has created ECS-A with security group Sys-A associated. – Principal B has created database RDS-B with security group Sys-B associated. The owner can add rules with **Source** set to **Sys-A** or **Sys-B** to security group **Sys-X**.	• Can create their own security groups. • Only has the operation permissions on their own security groups and cannot perform any operations on the security groups of the owner or other principals. • For security groups associated with resources in a shared subnet, a principal can add rules to their own security groups and can set **Source** of the rules to the security groups of the owner or other principals. For example, in the shared Subnet-X: – The owner has created ECS-X with security group Sys-X associated. – Principal A has created ECS-A with security group Sys-A associated. – Principal B has created database RDS-B with security group Sys-B associated. Principal A can add rules with **Source** set to **Sys-X** or **Sys-B** to security group **Sys-A**.
IP address group	IP address groups are independent of each other. Owners can create an IP address group and associate it with their own security groups.	IP address groups are independent of each other. Principals can create an IP address group and associate it with their own security groups.

Resource	Owner	Principal
VPC flow log	• Can create a flow log with **Resource Type** set to **VPC** or **Subnet**. Traffic on all network interfaces of the principal in the shared subnet will be recorded in this flow log. • Can create a flow log with **Resource Type** set to **NIC**. Traffic on all network interfaces of the owner will be recorded in this flow log.	Can create a flow log with **Resource Type** set to **NIC**. Traffic on all network interfaces of the principal will be recorded in this flow log.
VPC peering connection	Selects the VPC with subnets shared with other accounts to create a VPC peering connection.	Cannot select the VPC with subnets shared with other accounts to create a VPC peering connection.
NAT gateway	Creates and manages NAT gateways in the shared subnet.	Cannot create NAT gateways in the shared subnet.
VPN gateway	Creates and manages VPN gateways in the shared subnet.	Cannot create VPN gateways in the shared subnet.
Enterprise router	Attaches the VPC with subnets shared with other accounts to an enterprise router.	Cannot attach the VPC with subnets shared with other accounts to an enterprise router.
Enterprise switch	Creates and manages enterprise switches in the shared subnet.	Cannot create enterprise switches in the shared subnet.
Direct Connect connection	Creates and manages Direct Connect connections in the shared subnet.	Cannot create Direct Connect connections in the shared subnet.
Cloud connection	Loads the VPC with subnets shared with other accounts to a cloud connection.	Cannot load the VPC with subnets shared with other accounts to a cloud connection.
VPC endpoint	Creates and manages VPC endpoints in the shared subnet.	Cannot create VPC endpoints in the shared subnet.
Tag	Adds and manages tags in the shared subnet.	Cannot add tags in the shared subnet.

Billing

You only need to pay for the resources (such as ECSs, load balancers, and RDS instances) you create in the shared subnets. For details, see the billing description of each cloud resource.

Quotas

Table 9-4 lists the quotas of shared subnets. The quotas cannot be increased.

Table 9-4 Quotas

Item	Default Quota
Maximum number of subnet shares that a principal can receive	100
Maximum number of principal that a subnet can be shared with	100

Notes and Constraints

- A principal can receive a maximum of 100 subnet shares.
- A subnet can be shared with a maximum of 100 principals.
- The following cloud resources can be created in a shared subnet:
  - **ECSs**
  - **BMSs**
  - **Dedicated load balancers**
  - **CCE turbo clusters**
  - **API gateways**
  - **Kafka instances**
  - **ServiceStage environments**
  - **ServiceComb engines**
  - **FunctionGraph functions**
  - **DCS instances**
  - **GaussDB instances**
  - **TaurusDB**
  - **GeminiDB Influx instances**
  - **GeminiDB Redis instances**
  - **GeminiDB Cassandra instances**
  - **RDS for MySQL instances**
  - **RDS for PostgreSQL instances**
  - **DDS cluster instances**

- Dedicated HSM instances
- Database audit instances
- CBH instances
- GaussDB(DWS) instances
- DataArts Studio instances
- CSS clusters
- Network connections between DLI and resources
- CDM clusters
- Workspace

9.2 Usage Examples for VPC Sharing

Suppose you have two types of workloads running on the cloud. One type of workloads needs to access the Internet and the other type does not. To make resource management easier, you can use account A to manage basic, public IT resources, such as VPCs, subnets, and route tables. And you can share subnets in a VPC in account A with accounts B, C, and D, so the principals can create resources, such as ECSs, RDS instances, and load balancers, in the shared subnets. You can plan your VPC sharing by referring to **Figure 9-3**, and plan accounts and resources as described in **Table 9-5**.

Figure 9-3 Planning on VPC sharing

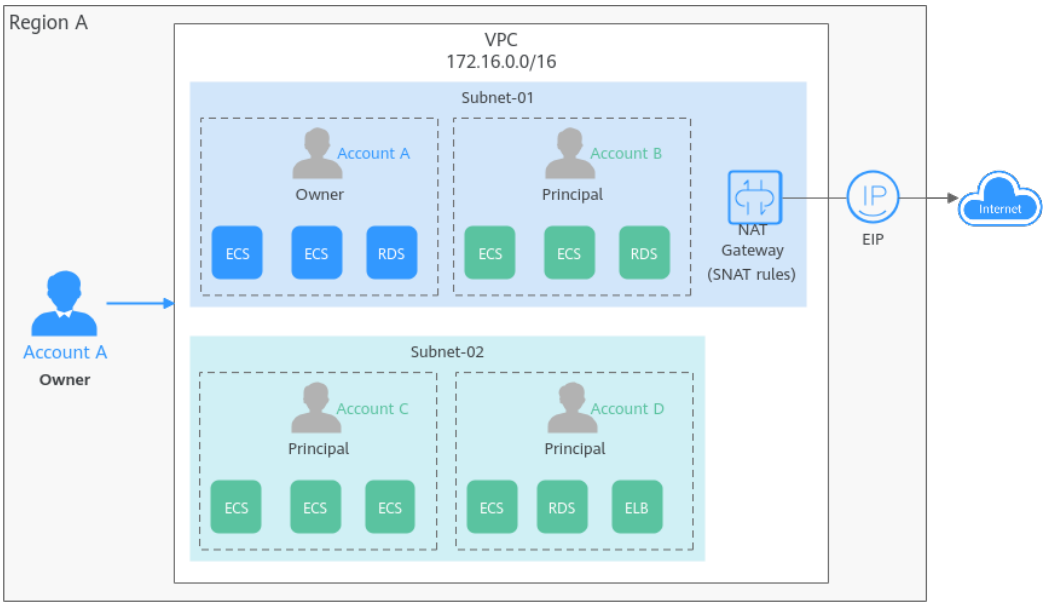


Table 9-5 Planning on VPC sharing

Account	Role	Resource Permissions
Account A	Owner	• Creates a VPC and subnets and shares the subnets with other accounts. • Creates a NAT gateway with an EIP bound and configures SNAT rules to enable Subnet-01 to connect to the Internet.
Account B	Principal	Creates ECSs and RDS instances in Subnet-01 to deploy applications that can be accessed over the Internet.
Accounts C and D	Principal	Create ECSs, RDS instances, and load balancers in Subnet-02. These resources do not need to connect to the Internet.

Subnets in the same VPC can communicate with each other by default. However, if instances are associated with different security groups, they are isolated from each other. If you want the resources to communicate with each other, you need to add security group rules to allow the communications.

- Resources in account A are protected by security group Sg-A.
- Resources in account B are protected by security group Sg-B.
- Resources in account C are protected by security group Sg-C.
- Resources in account D are protected by security group Sg-D.

To enable resources in accounts C and D to communicate with each other, add inbound rules to security groups Sg-C and Sg-D.

Table 9-6 Inbound rules

Security Group	Direction	Priority	Action	Type	Protocol & Port	Source
Sg-C	Inbound	1	Allow	IPv4	Specify the protocol and port as needed. Example: **Protocol/All**	Security group Sg-D

Security Group	Direction	Priority	Action	Type	Protocol & Port	Source
Sg-D	Inbound	1	Allow	IPv4	Specify the protocol and port as needed. Example: **Protocol/All**	Security group Sg-C

9.3 Sharing a Subnet with Other Accounts

Scenarios

The owner of a VPC can share subnets in the VPC with principals. You need to create a resource share, select the subnets to share, associate permissions, and specify principals. Once a subnet is shared, principals can create instances in this subnet.

Prerequisites

A resource share is available. If there are no resource shares, create one by referring to **Creating a Resource Share**.

Notes and Constraints

- A principal can receive a maximum of 100 subnet shares.
- A subnet can be shared with a maximum of 100 principals.

Procedure

1. Log in to the management console.
2. Click in the upper left corner and select the desired region and project.
3. Click in the upper left corner and choose **Networking** > **Virtual Private Cloud**.

   The **Virtual Private Cloud** page is displayed.
4. Locate the subnet to share and click its name.

   The **Summary** page is displayed.
5. Click the **Sharing** tab and click **Share Subnet**.

   The **Share Subnet** dialog box is displayed.
6. Select an available resource share.

   If there is no resource share available, create one.

a. Click **Cancel** to close the **Share Subnet** dialog box.

   The **Sharing** tab is displayed.

b. Click **Create Resource Share**.

   Create a resource share on the RAM console by referring to **Creating a Resource Share**.

c. Repeat **5** to **6** to add subnets to the existing resource share.

7. Click **OK**.

   Return to the **Sharing** tab. You can view that the resource share is in **Sharing** status.

Follow-up Operations

After an owner shares a subnet with a principal, the principal needs to accept the sharing within a specified period to use the subnet. For details, see **Responding to a Resource Sharing Invitation**.

9.4 Viewing the Details of a Shared Subnet

Scenarios

The owner and principals of a shared subnet can view the details about a shared subnet, such as the name and status of the shared subnet.

Procedure

1. Log in to the management console.
2. Click  in the upper left corner and select the desired region and project.
3. Click  in the upper left corner and choose **Networking** > **Virtual Private Cloud**.

   The **Virtual Private Cloud** page is displayed.
4. Locate the target subnet and click its name.

   The **Summary** page is displayed.
5. Click the **Sharing** tab and view the name and status of the resource share that the subnet belongs to.
   - If you are the owner of a shared subnet, you can view shared resources, permissions, and principals on the RAM management console. For details, see **Viewing a Resource Share**.
   - If you are a principal of a shared subnet, you can view shared resources, permissions, and resource owner on the RAM management console. For details, see **Viewing Resources Shared with You**.

9.5 Stopping Sharing a Subnet

Scenarios

The owner of a shared subnet can stop sharing a subnet. After the share is stopped, principals cannot create resources in the subnet, but existing resources can be used normally.

Procedure

1. Log in to the management console.
2. Click  in the upper left corner and select the desired region and project.
3. Click  in the upper left corner and choose **Networking** > **Virtual Private Cloud**.

   The **Virtual Private Cloud** page is displayed.
4. Locate the subnet to share and click its name.

   The **Summary** page is displayed.
5. Click the **Sharing** tab, locate the row that contains the resource share, and click **Stop Sharing** in the **Operation** column.

   A confirmation dialog box is displayed.
6. Confirm the information and click **OK**.

   Return to the **Sharing** tab page. You can view that the resource share is in the **Sharing stopped** status.

10 Edge Gateway

10.1 Edge Gateway Overview

What Is an Edge Gateway?

An edge gateway can connect subnets in the same VPC but from both edge and central AZs or from different edge AZs.

NOTE

Edge gateways are supported in the AF-Johannesburg region.

Edge gateways are free of charge, but if there is a change, you will be notified in advance.

Application Scenarios

Figure 10-1 Edge gateway networking architecture

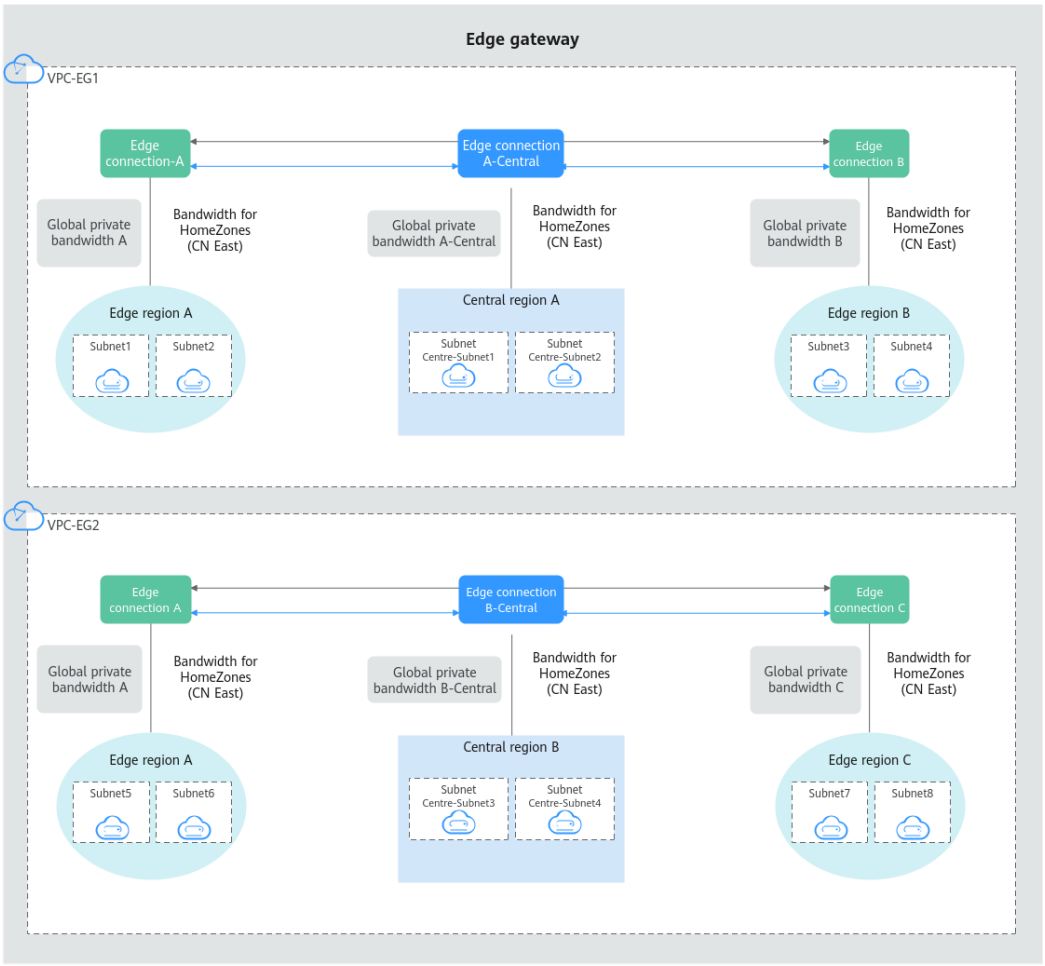


Table 10-1 lists the required two VPCs and 12 subnets in 12 AZs and five regions.

Table 10-1 VPC and subnet planning

VPC	Subnet	AZ	Region
VPC-EG1	Subnet1	Edge AZ1	Edge region A
	Subnet2	Edge AZ2	
	Subnet3	Edge AZ3	Edge region B
	Subnet4	Edge AZ4	
	Centre-Subnet1	Central AZ1	Central region A
	Centre-Subnet2	Central AZ2	
VPC-EG2	Subnet5	Edge AZ5	Edge region A
	Subnet6	Edge AZ6	

VPC	Subnet	AZ	Region
	Subnet7	Edge AZ7	Edge region C
	Subnet8	Edge AZ8	
	Centre-Subnet3	Central AZ3	Central region B
	Centre-Subnet4	Central AZ4	

Table 10-2 lists the required edge gateway and five edge connections in 12 AZs and five regions.

Table 10-2 Edge gateway and edge connection planning

Edge Gateway	AZ	Region	Edge Connection	Global Connection Bandwidth-HomeZones (CN East)
Edge gateway	• Edge AZ1 • Edge AZ2 • Edge AZ5 • Edge AZ6	Edge region A	Edge connection A	Global connection bandwidth A
	• Edge AZ3 • Edge AZ4	Edge region B	Edge connection B	Global connection bandwidth B
	• Edge AZ7 • Edge AZ8	Edge region C	Edge connection C	Global connection bandwidth C
	• Central AZ1 • Central AZ2	Central region A	Edge connection A-Central	Global connection bandwidth A-Central
	• Central AZ3 • Central AZ4	Central region B	Edge connection B-Central	Global connection bandwidth B-Central

The edge gateway is attached to both **VPC-EG1** and **VPC-EG2** and is used in the following scenarios.

Table 10-3 Edge gateway application scenarios

Scenario	Required Resource	Description	Reference
In **VPC-EG1**, connect subnets in edge and central AZs.	• VPC • Subnet in an edge AZ • Edge connection in an edge region • Global connection bandwidth in an edge region • Subnet in a central AZ • Edge connection in a central region • Global connection bandwidth in a central region	In **VPC-EG1**, **Subnet1** in the edge AZ (**AZ1**) and **Centre-Subnet1** in the central AZ (**AZ1**) need to communicate with each other. • For the subnet in the edge AZ (**AZ1**), an edge connection (**edge connection A**) needs to be created in edge region A and has a global connection bandwidth (**global connection bandwidth A**) bound. • For the subnet in the central AZ (**AZ1**), an edge connection (**edge connection A-Central**) needs to be created in central region A and has a global connection bandwidth (**global connection bandwidth A-Central**) bound.	1. Edge gateways: An edge gateway can have one or more VPCs attached. 2. **Associating VPCs with an Edge Gateway**: The number of VPCs that an edge gateway can have vary by gateway edition. 3. **Creating an Edge Connection**: An edge connection can be shared in all AZs of the same region. 4. **Binding a Global Connection Bandwidth to an Edge Connection**: An edge connection can only have one global connection bandwidth bound.

Scenario	Required Resource	Description	Reference
In **VPC-EG1**, connect subnets in different edge AZs.	• VPC • Subnet in an edge AZ • Edge connection in an edge region • Global connection bandwidth in an edge region • Edge connection in a central region • Global connection bandwidth in a central region	In **VPC-EG1**, **Subnet1** in the edge AZ (**AZ1**) and **Subnet3** in the edge AZ (**AZ3**) need to communicate with each other. • For the subnet in the edge AZ (**AZ1**), an edge connection (**edge connection A**) needs to be created in edge region A and has a global connection bandwidth (**global connection bandwidth A**) bound. • For the subnet in the edge AZ (**AZ3**), an edge connection (**edge connection B**) needs to be created in edge region B and has a global connection bandwidth (**global connection bandwidth B**) bound. • For both subnets, an edge connection (**edge connection A-Central**) needs to be created in central region A and has a global connection bandwidth (**global connection bandwidth A-Central**) bound. **NOTE** To enable communications between a central region and an edge region or between different edge regions, you need to create an edge connection in the central region.	1. Edge gateways: An edge gateway can have one or more VPCs attached. 2. **Associating VPCs with an Edge Gateway**: The number of VPCs that an edge gateway can have vary by gateway edition. 3. **Creating an Edge Connection**: This edge connection is used for communications between the edge regions and a central region. 4. **Binding a Global Connection Bandwidth to an Edge Connection**: An edge connection can only have one global connection bandwidth bound.

Scenario	Required Resource	Description	Reference
In **VPC-EG2**, connect subnets in edge and central AZs.	• VPC • Subnet in an edge AZ • Edge connection in an edge region • Global connection bandwidth in an edge region • Subnet in a central AZ • Edge connection in a central region • Global connection bandwidth in a central region	In **VPC-EG2**, **Subnet5** in the edge AZ (**AZ5**) and **Centre-Subnet3** in the central AZ (**AZ3**) need to communicate with each other. • For the subnet in the edge AZ (**AZ5**), an edge connection has been created in edge region A and has a global connection bandwidth bound. You can use this edge connection directly. • For the subnet in the central AZ (**AZ3**), an edge connection (**edge connection B-Central**) needs to be created in central region B and has a global connection bandwidth (**global connection bandwidth B-Central**) bound. **NOTE** Edge connections can be reused because **VPC-EG1** and **VPC-EG2** are attached to the same edge gateway.	1. Edge gateways: An edge gateway can have one or more VPCs attached. 2. **Associating VPCs with an Edge Gateway**: The number of VPCs that an edge gateway can have vary by gateway edition. 3. **Creating an Edge Connection**: VPCs attached to the same edge gateway can use one edge connection. In this example, only one edge connection needs to be created in the central region. 4. **Binding a Global Connection Bandwidth to an Edge Connection**: An edge connection can only have one global connection bandwidth bound.

Scenario	Required Resource	Description	Reference
In **VPC-EG2**, connect subnets in different edge regions.	• VPC • Subnet in an edge AZ • Edge connection in an edge region • Global connection bandwidth in an edge region • Edge connection in a central region • Global connection bandwidth in a central region	In **VPC-EG2**, **Subnet5** in the edge AZ (**AZ5**) and the subnet in the edge AZ (**AZ7**) need to communicate with each other. • For the subnet in the edge AZ (**AZ5**), an edge connection has been created in edge region A and has a global connection bandwidth bound. You can use this edge connection directly. • For the subnet in the edge AZ (**AZ7**), an edge connection (**edge connection C**) needs to be created in edge region C and has a global connection bandwidth (**global connection bandwidth C**) bound. • For both subnets, an edge connection (**edge connection B-Central**) needs to be created in central region B and has a global connection bandwidth (**global connection bandwidth B-Central**) bound.	1. Edge gateways: An edge gateway can have one or more VPCs attached. 2. **Associating VPCs with an Edge Gateway**: The number of VPCs that an edge gateway can have vary by gateway edition. 3. **Creating an Edge Connection**: VPCs attached to the same edge gateway can use one edge connection. In this example, one edge connection is required for the edge AZ (**AZ7**), and one edge connection needs to be created in the central region. 4. **Binding a Global Connection Bandwidth to an Edge Connection**: An edge connection can only have one global connection bandwidth bound.

10.2 Buying an Edge Gateway

Scenarios

This section describes how to buy an edge gateway. Edge gateways and global connection bandwidths can work together to allow resources from the central and edge subnets, and from different edge subnets in VPCs to communicate with each other through an internal network.

NOTE

Edge gateways are supported in the AF-Johannesburg region.

Edge gateways are free of charge, but if there is a change, you will be notified in advance.

Procedure

1. Go to the **edge gateway list page**.
2. In the upper right corner of the page, click **Buy Edge Gateway**.
   The **Buy Edge Gateway** page is displayed.
3. Configure the parameters based on **Table 10-4**.

Table 10-4 Parameter descriptions

Parameter	Description	Example Value
Billing Mode	Mandatory **Pay-per-use**: a postpaid subscription. You pay for the edge gateway based on the amount of time you use the edge gateway. Your edge gateway is billed by the second, and settled by the hour. If the usage is less than an hour, you are billed based on the actual duration consumed. Edge gateways are free of charge, but if there is a change, you will be notified in advance.	Pay-per-use
Region	Mandatory Regions are geographic areas isolated from each other. Resources are region-specific and cannot be used across regions through internal network connections. For low network latency and quick resource access, select the nearest region.	-
Name	Mandatory Enter the name of the bandwidth. The name: • Can contain 1 to 64 characters. • Can contain letters, digits, underscores (_), hyphens (-), and periods (.).	-

Parameter	Description	Example Value
Edition	Mandatory Select the edition of an edge gateway. • **Basic**: A maximum of 2 VPCs can be associated with a basic edge gateway. • **Enterprise**: A maximum of 10 VPCs can be associated with an enterprise edge gateway. • **Enterprise pro**: A maximum of 30 VPCs can be associated with an enterprise-pro edge gateway. The edition of an edge gateway can be changed after it is purchased. For details, see **Changing the Edition of an Edge Gateway**.	Basic
VPC	Mandatory Associate one or more VPCs with an edge gateway. You can associate VPCs with an edge gateway to allow resources from the central and edge subnets, and from different edge subnets in these VPCs to communicate with each other through an internal network. After an edge gateway is purchased, you can **associate VPCs with this gateway** or **disassociate VPCs from this gateway**.	vpc-test01 vpc-test02
Advanced Settings/ Description	Optional Enter the description of the edge gateway in the text box as required.	-
Advanced Settings/Tag	Optional You can add tags to the edge gateway. Tags help you to identify, classify, and search for cloud resources. For details, see **Managing the Tags of an Edge Gateway**.	**Tag key**: test **Tag value**: 01

4. Click **Next**.
5. Confirm the edge gateway information and click **Submit**.
   The **Edge Gateways** page is displayed.
6. Check the status of the edge gateway.
   If the status is **Running**, the edge gateway is purchased.

Follow-up Operations

- Mandatory: Associate VPCs with an edge gateway. An edge gateway cannot work independently. For details, see **Associating VPCs with an Edge Gateway**.
- Mandatory: Create edge connections for an edge gateway to enable resources from the central and edge subnets, and from different edge subnets to communicate with each other through an internal network. For details, see **Creating an Edge Connection**.
- Mandatory: Bind a global connection bandwidth to each edge connection to enable communications through an internal network. For details, see **Binding a Global Connection Bandwidth to an Edge Connection**.

10.3 Associating VPCs with or Disassociating VPCs from an Edge Gateway

Scenarios

This section describes how to associate VPCs with or disassociate VPCs from an edge gateway.

- **Associating VPCs with an Edge Gateway**: You can associate one or more VPCs with an edge gateway at a time. The number of VPCs that can be associated with an edge gateway depends on the gateway edition:
  - **Basic**: A maximum of 2 VPCs can be associated with a basic edge gateway.
  - **Enterprise**: A maximum of 10 VPCs can be associated with an enterprise edge gateway.
  - **Enterprise pro**: A maximum of 30 VPCs can be associated with an enterprise-pro edge gateway.
- **Disassociating VPCs from an Edge Gateway**: You can disassociate one or more VPCs from an edge gateway at a time. Ensure that there are no services running in these VPCs before proceeding with the disassociation.

Associating VPCs with an Edge Gateway

1. Go to the **edge gateway list page**.
2. In the edge gateway list, search for or locate the edge gateway.
3. Click the edge gateway name.

   The **Basic Information** tab is displayed.
4. Click the **VPCs** tab and click **Associate**.

   The **Associate VPC** dialog box is displayed.
5. In the VPC list, select one or more VPCs and click **OK**.

   On the **VPCs** tab, you can view the VPCs associated with the edge gateway.

Disassociating VPCs from an Edge Gateway

1. Go to the **edge gateway list page**.

2. In the edge gateway list, search for or locate the edge gateway.
3. Click the edge gateway name.

   The **Basic Information** tab is displayed.
4. Click the **VPCs** tab. In the VPC list, disassociate VPCs.
   - Disassociating a single VPC:
     i. In the VPC list, locate the target VPC and click **Disassociate** in the **Operation** column.

        A confirmation dialog box is displayed.
     ii. Confirm the information and click **OK**.

        The disassociated VPCs are not displayed in the VPC list.
   - Disassociating multiple VPCs:
     i. In the VPC list, select the VPCs to be disassociated and click **Disassociate** in the upper left corner of the VPC list.

        A confirmation dialog box is displayed.
     ii. Confirm the information and click **OK**.

        The disassociated VPCs are not displayed in the VPC list.

10.4 Managing Edge Gateways

Scenarios

This section describes how to manage your edge gateways.

- **Changing the Edition of an Edge Gateway**
- **Viewing an Edge Gateway**
- **Deleting an Edge Gateway**

Constraints

- If you want to delete an edge gateway with VPCs associated, disassociate the VPCs first. For details, see **Disassociating VPCs from an Edge Gateway**.
- If you want to delete an edge gateway with edge connections, delete the edge connections first. For details, see **Deleting an Edge Connection**.

Changing the Edition of an Edge Gateway

1. Go to the **edge gateway list page**.
2. In the edge gateway list, search for or locate the edge gateway.
3. Locate the target edge gateway and click **Change Edition** in the **Operation** column.

   The **Change Edition** page is displayed.
4. Select an edge gateway edition as prompted.

   Select the edition of an edge gateway.
   - **Basic**: A maximum of 2 VPCs can be associated with a basic edge gateway.

- **Enterprise**: A maximum of 10 VPCs can be associated with an enterprise edge gateway.
- **Enterprise pro**: A maximum of 30 VPCs can be associated with an enterprise-pro edge gateway.

5. Click **Next**.
6. Confirm the edge gateway information and click **Submit**.

   The **Edge Gateways** page is displayed and you can view that the edition of the edge gateway was changed.

Viewing an Edge Gateway

1. Go to the **edge gateway list page**.
2. In the edge gateway list, search for or locate the edge gateway.
3. Click the edge gateway name.

   On the **Basic Information** tab, view more details.

Deleting an Edge Gateway

1. Go to the **edge gateway list page**.
2. In the edge gateway list, search for or locate the edge gateway.
3. Locate the target edge gateway you want to delete and click **Delete** in the **Operation** column.

   A confirmation dialog box is displayed.
4. Confirm the information and click **OK**.

   The deleted edge gateway is not displayed in the gateway list.

10.5 Managing the Tags of an Edge Gateway

Scenarios

Tags help you identify, classify, and search for edge gateways. This section shows you how to:

- Add a tag to an edge gateway.
- Modify an edge gateway tag.
- Delete a tag from an edge gateway.

For details about the edge gateway tag requirements, see **Table 10-5**.

Table 10-5 Tag key and value requirements

Parameter	Requirements	Example Value
Tag key	• For each resource, each tag key must be unique, and each tag key can only have one tag value. • Cannot be left blank. • Can contain a maximum of 128 characters. • Can contain letters in any language, digits, spaces, underscores (_), periods (.), colons (:), equal signs (=), plus signs (+), minus signs (-), and at signs (@). • Cannot start with _sys_ or a space or end with a space.	test
Tag value	• Can be left blank. • Can contain a maximum of 255 characters. • Can contain letters in any language, digits, spaces, underscores (_), periods (.), colons (:), slashes (/), equal signs (=), plus signs (+), minus signs (-), and at signs (@). • Cannot start or end with a space.	01

Notes and Constraints

Each cloud resource can have a maximum of 20 tags.

Procedure

1. Go to the **edge gateway list page**.
2. On the **Tags** tab, click **Edit Tag** in the upper left corner above the tag list.
   The **Edit Tag** dialog box is displayed.
3. Perform the following operations on the tag as required:
   - Adding a tag: Click +, enter a tag key and value, and click **OK**.
   - Modifying a tag: Click × next to the target tag key or value, delete the original value, enter a new value, and click **OK**.
   - Deleting a tag: Click **Delete** next to the target tag and click **OK**.

10.6 Creating an Edge Connection

Scenarios

This section describes how to create an edge connection. Edge connections allow subnets from central and edge AZs, and from different edge AZs in VPCs to communicate with each other through an internal network.

Constraints

- An edge connection needs to have a global connection bandwidth bound to enable resources from the central and edge sites, and from different edge sites to communicate with each other.
- If you want to enable resources from the central and edge regions, and from different edge regions to communicate, you need to create edge connections in the central region. You can create edge connections in edge regions based on actual requirements.

  For example, Subnet1 in edge AZ1 and Subnet3 in edge AZ3 are from the same VPC-EG1 but different regions. If you want resources from the two subnets to communicate, you need to create an edge connection in central AZ1, edge AZ1, and edge AZ3, respectively.
- Edge connections can be reused if different VPCs are associated with the same edge gateway.

  For example, an edge gateway is associated with both VPC-EG1 and VPC-EG2, and edge AZ1 and edge AZ5 are both in edge region A. If you need Subnet5 in edge AZ5 to communicate with Centre-Subnet3 in central AZ3, you can directly use the created edge connection-A in edge AZ1 of edge region A. You only need to create an edge connection for the region of the central AZ3 and bind a global connection bandwidth.

Table 10-6 Resource configuration examples

Edge Gateway	VPC	Subnet	AZ	Region	Edge Connection	Global Connection Bandwidth-HomeZones (CN East)
Edge gateway	VPC-EG1	Subnet1	Edge AZ1	Edge region A	Edge connection A	Global connection bandwidth A
		Subnet3	Edge AZ3	Edge region B	Edge connection B	Global connection bandwidth B
		Centre-Subnet1	Central AZ1	Central region A	Edge connection -Central A	Global connection bandwidth-Central A

Edge Gateway	VPC	Subnet	AZ	Region	Edge Connection	Global Connection Bandwidth-HomeZones (CN East)
	VPC-EG2	Subnet5	Edge AZ5	Edge region A	Edge connection A	Global connection bandwidth A
		Centre-Subnet3	Central AZ3	Central region B	Edge connection -Central B	Global connection bandwidth-Central B

Procedure

1. Go to the **edge gateway list page**.
2. Click the edge gateway name.
   The **Basic Information** tab is displayed.
3. Click the **Edge Connections** tab, and then click **Create Edge Connection**.
   The **Create Edge Connection** dialog box is displayed.
4. Configure parameters shown in **Table 10-7** for the edge connection as prompted.

Table 10-7 Parameter descriptions

Parameter	Description	Example Value
Name	Mandatory Enter an edge connection name. The name: • Can contain 1 to 64 characters. • Can contain letters, digits, underscores (_), hyphens (-), and periods (.).	-
AZ	Mandatory Select an AZ to create an edge connection. Only one edge connection can be created for the AZs in the same network border group. AZs in the same network border group are at the same edge site. You can select any AZ at the same edge site.	-
Description	Optional Enter a description for the edge connection in the text box as required.	-

5. After the parameters are configured, click **OK**.

Follow-up Operations

You need bind a global connection bandwidth to each edge connection to enable communications through an internal network. For details, see **Binding a Global Connection Bandwidth to an Edge Connection**.

10.7 Binding or Unbinding a Global Connection Bandwidth to and from an Edge Connection

Scenarios

This section describes how to bind or unbind a global connection bandwidth to or from an edge connection.

- **Binding a Global Connection Bandwidth to an Edge Connection**: Bind a global connection bandwidth to an edge connection to enable resources from the central and edge subnets, and from different edge subnets to communicate with each other.
- **Unbinding a Global Connection Bandwidth from an Edge Connection**: Ensure that no services are running on an edge connection before unbinding its global connection bandwidth.

In addition, you can modify the global connection bandwidth name and size by referring to **Modifying the Global Connection Bandwidth of an Edge Connection**.

Binding a Global Connection Bandwidth to an Edge Connection

1. Go to the **edge gateway list page**.
2. Click the edge gateway name.

   The **Basic Information** tab is displayed.
3. Click the **Edge Connections** tab. In the edge connection list, search for or locate the edge connection.
4. Click **Bind Global Connection Bandwidth** on the right of the target edge connection.

   A confirmation dialog box is displayed.
5. Select a global connection bandwidth as prompted and click **OK**.

   NOTE

   If there is no global connection bandwidth, you can purchase one by referring to .**Buying a Global Connection Bandwidth**
6. Check whether the binding is successful. If the **Bind Global Connection Bandwidth** button on the right of the target edge connection turns gray, the binding is successful.

Unbinding a Global Connection Bandwidth from an Edge Connection

1. Go to the **edge gateway list page**.
2. Click the edge gateway name.

   The **Basic Information** tab is displayed.
3. Click the **Edge Connections** tab. In the edge connection list, search for or locate the edge connection.
4. Locate the target edge connection and click **More** > **Unbind Global Connection Bandwidth** in the **Operation** column.

   A confirmation dialog box is displayed.
5. Confirm the information and click **OK**.
6. Check whether the unbinding is successful. If the **Bind Global Private Bandwidth** on the right of the target edge connection turns black, the unbinding is successful.

Modifying the Global Connection Bandwidth of an Edge Connection

You can modify the global connection bandwidth name and size.

For details, see **Modifying a Global Connection Bandwidth**.

10.8 Managing Edge Connections

Scenarios

This section describes how to manage your edge connections.

- **Viewing an Edge Connection**
- **Deleting an Edge Connection**

Constraints

If you want to delete an edge connection with a global connection bandwidth bound, unbind the global connection bandwidth first. For details, see **Unbinding a Global Connection Bandwidth from an Edge Connection**.

Viewing an Edge Connection

1. Go to the **edge gateway list page**.
2. Click the edge gateway name.

   The **Basic Information** tab is displayed.
3. Click the **Edge Connections** tab. In the edge connection list, you can view the AZ and the bound global connection bandwidth.

Deleting an Edge Connection

1. Go to the **edge gateway list page**.
2. Click the edge gateway name.

   The **Basic Information** tab is displayed.

3. Click the **Edge Connections** tab. Locate the target edge connection you want to delete and choose **More** > **Delete** in the **Operation** column.

   A confirmation dialog box is displayed.

4. Confirm the information and click **OK**.

   The deleted edge connection is not displayed in the connection list.

11 IPv4/IPv6 Dual-Stack Network

What Is an IPv4/IPv6 Dual-Stack Network?

An IPv4/IPv6 dual-stack network allows your resources, such as ECSs, to use both IPv4 and IPv6 addresses for private and public network communications. **Figure 11-1** shows how an IPv4/IPv6 dual-stack network works.

Figure 11-1 An IPv4/IPv6 dual-stack network

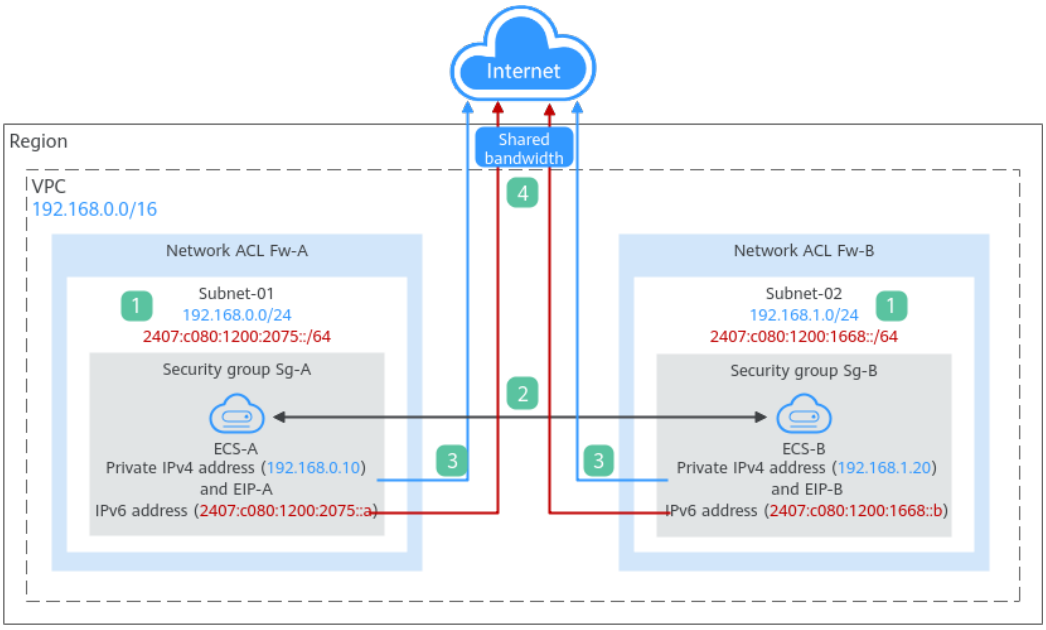


Table 11-1 Steps for deploying a dual-stack network

Step	Description
1	If you enable IPv6 when adding a VPC subnet, an IPv6 CIDR block is automatically assigned to the subnet. You cannot customize the IPv6 CIDR block.

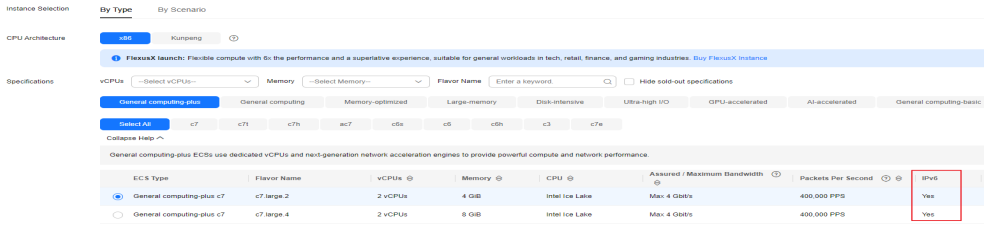
Step	Description
2	Subnets in the same VPC can communicate with each other by default. Network ACLs protect subnets, and security groups protect the instances in it. 1. Subnets in different network ACLs are isolated from each other. To connect these subnets, you need to add inbound and outbound rules to allow traffic in and out of the subnets. 2. Security groups are isolated from each other. If two instances are associated with different security groups, you need to add inbound and outbound rules to allow the instances to communicate with each other. As shown in **Figure 11-1**, if allow rules are configured for network ACLs **Fw-A** and **Fw-B** and security groups **Sg-A** and **Sg-B**, **ECS-A** and **ECS-B** can communicate with each other: • Using private IPv4 addresses (**192.168.0.10** and **192.168.1.20**). • Using IPv6 addresses (**2407:c080:1200:2075::a** and **2407:c080:1200:1668::b**).
3	To enable instances to communicate with the Internet using IPv4 addresses, you need to buy an EIP and bind it to the instance. An EIP can be bound to only one instance. As shown in **Figure 11-1**, you can bind **EIP-A** to **ECS-A** and **EIP-B** to **ECS-B** so that **ECS-A** and **ECS-B** can communicate with the Internet.
4	To enable instances to communicate with the Internet using an IPv6 address, you need to add the IPv6 address to a shared bandwidth. You can add multiple IPv6 addresses to a shared bandwidth. As shown in **Figure 11-1**, you can add the IPv6 addresses of **ECS-A** and **ECS-B** to a shared bandwidth so that **ECS-A** and **ECS-B** can communicate with the Internet.

Notes and Constraints

- The IPv4/IPv6 dual-stack function is free for now, but will be billed at a later date (price yet to be determined).
- The IPv6 function is now available for open beta test in **certain regions**. You can use the IPv6 function only after obtaining the OBT permission.
- Only certain ECS flavors support IPv6 networks. You need to select such ECSs in supported regions.

  On the ECS console, click **Buy ECS**. On the displayed page, check the ECS specifications. If **Yes** is shown in the **IPv6** column, the ECS with this specification supports IPv6.

Figure 11-2 ECS specifications



IPv4/IPv6 Dual-Stack Application Scenarios

If your ECS supports IPv6, you can build an IPv4/IPv6 dual-stack network. **Table 11-2** shows where IPv4/IPv6 dual-stack networks can be used.

Table 11-2 Application scenarios of IPv4/IPv6 dual-stack networks

Application Scenario	Scenario	Subnet	ECS
Private communication using IPv6 addresses	Your applications deployed on ECSs need to communicate with other systems (such as databases) through private networks using IPv6 addresses.	• IPv4 CIDR block • IPv6 CIDR block	• Private IPv4 address: used for private communication • IPv6 address: used for private communication.
Public communication using IPv6 addresses	Your applications deployed on ECSs need to provide services accessible from the Internet using IPv6 addresses.	• IPv4 CIDR block • IPv6 CIDR block	• Private IPv4 address + IPv4 EIP: used for public network communication • IPv6 address + shared bandwidth: used for public network communication
	Your applications deployed on ECSs need to both provide services accessible from the Internet and analyze the access request data using IPv6 addresses.		

If your ECS flavor does not support IPv6 addresses, you can enable the IPv6 EIP function to allow communications using IPv6 addresses. For details, see **Table 11-3**.

Table 11-3 Application scenarios of IPv6 EIPs

Application Scenario	Description	Subnet	ECS
Public communication using IPv6 addresses	Your applications deployed on ECSs need to provide services accessible from the Internet using IPv6 addresses.	IPv4 CIDR block	• Private IPv4 address • IPv4 EIP (with IPv6 function enabled): used for public communication using IPv4 and IPv6 EIPs

Figure 11-3 Application scenarios of IPv6 networks

Operation Guide on IPv6 Networks

Operations on an IPv6 network are similar to those on an IPv4 network. Only some functions are configured in a different way. **Table 11-4** describes how you can build and use an IPv6 network.

Table 11-4 Operation guide on IPv6 networks

Scenario	Description	Reference
Creating an IPv6 subnet	Select **Enable** for **IPv6 CIDR Block** when creating a subnet. An IPv6 CIDR block will be automatically assigned to the subnet. • You cannot customize an IPv6 CIDR block. • IPv6 cannot be disabled after the subnet is created. • You can enable IPv6 for existing subnets.	**Creating a Subnet for an Existing VPC**
Viewing in-use IPv6 addresses	In the subnet list, click the subnet name. On the displayed page, view in-use IPv4 and IPv6 addresses on the **IP Addresses** tab.	**Viewing IP Addresses in a Subnet**
Adding a security group rule (IPv6)	Add a security group rule with **Type** set to **IPv6** and **Source** or **Destination** set to an IPv6 address or IPv6 CIDR block.	**Adding a Security Group Rule**
Adding a network ACL rule (IPv6)	Add a network ACL rule with **Type** set to **IPv6** and **Source** or **Destination** set to an IPv6 address or IPv6 CIDR block.	**Adding a Network ACL Rule**
Purchasing an EIP (IPv6)	When purchasing an EIP, select **Enable IPv6 Internet access**, or choose **More** > **Enable IPv6 EIP** in the **Operation** column of an existing IPv4 EIP. After IPv6 EIP is enabled, both IPv4 and IPv6 EIPs are assigned.	**IPv6 EIP**
Adding an IPv6 EIP or IPv6 address to a shared bandwidth	After purchasing a shared bandwidth, you can add IPv6 EIPs or IPv6 addresses to it.	**Adding an EIP to a Shared Bandwidth**
Adding an IPv6 route to the VPC route table	Add a route with **Destination** and **Next Hop** set to an IPv4 or IPv6 CIDR block. • If the destination is an IPv6 CIDR block, the next hop can only be an IP address in the same VPC as the IPv6 CIDR block. • If the destination is an IPv6 CIDR block, the next hop type can only be ECS, extension NIC, or virtual IP address. They must also have IPv6 addresses.	**Adding Routes to a Route Table**

Scenario	Description	Reference
Assigning a virtual IPv6 address	If IPv6 is enabled for a VPC subnet, you can set **IP Address Type** to **IPv6** when assigning for a virtual IP address.	**Assigning a Virtual IP Address**

12 VPC Flow Log

12.1 VPC Flow Log

VPC Flow Log

VPC flow logs help you collect traffic information about instances in a specified VPC, including inbound and outbound traffic. After creating a flow log, you can view the flow log records in the log group that you configured.

Flow logs can help you:

- Monitor the traffic of security groups and network ACL and optimize their rules.
- Monitor the traffic of network instances and analyze network attacks.
- Determine the direction of the traffic to and from network interfaces.

The collection of flow log data does not affect the throughput or latency of your network. You can create or delete flow logs as required, which does not affect your network performance.

NOTICE

Currently, the VPC flow log function is supported in certain regions. You can go to **Function Overview** and click **VPC Flow Log** to check.

The VPC flow log function itself is free of charge, but you may be charged for other resources used. For example, if data is stored in Log Tank Service (LTS), you will be billed based on the LTS standards. For details, see the ***Log Tank Service User Guide***.

VPC Flow Log Data

You can create a flow log for a network interface, subnet, or VPC. If you create a flow log for a subnet or a VPC, each network interface in the subnet or VPC is monitored.

The traffic of a monitored network interface is collected and flow log data is generated, including the network interface ID, source address, destination address, source port, destination port, and packet size of the traffic.

Table 12-1 VPC flow log field description

Field	Description	Example
version	VPC flow log version.	1
project-id	ID of the project that the object monitored by flow log belongs to.	5f67944957444bd6bb4fe3b367de8f3d
interface-id	ID of the network interface that the flow log data is generated for.	1d515d18-1b36-47dc-a983-bd6512aed4bd
srcaddr	Source address.	192.168.0.154
dstaddr	Destination address.	192.168.3.25
srcport	Source port.	38929
dstport	Destination port.	53
protocol	Internet Assigned Numbers Authority (IANA) protocol number. For details, see **Assigned Internet Protocol Numbers**.	17
packets	The number of packets transferred during the capture window.	1
bytes	The number of bytes transferred during the capture window.	96
start	The time, in Unix seconds, of the start of the capture window.	1548752136
end	The time, in Unix seconds, of the end of the capture window.	1548752736
action	The action that is associated with the traffic. • **ACCEPT**: The traffic was allowed by security groups or network ACLs. • **REJECT**: The traffic was denied by security groups or network ACLs.	ACCEPT

Field	Description	Example
log-status	The logging status of the VPC flow log. • **OK**: Data is logged normally to the chosen destinations. • **NODATA**: There was no traffic to or from the network interface during the capture window. • **SKIPDATA**: Some flow log records were skipped during the capture window. This may be caused by an internal capacity constraint or an internal error. Example: When **Filter** is set to **Accepted traffic**, if there is accepted traffic, the value of **log-status** is **OK**. If there is no accepted traffic, the value of **log-status** is **NODATA** regardless of whether there is rejected traffic. If some accepted traffic is abnormally skipped, the value of **log-status** is **SKIPDATA**.	OK

Notes and Constraints

- Currently, S2, M2, Hc2, H2, D2, P1, G3, Pi1, FP1, S3, C3, M3, H3, D3, Ir3, I3, Sn3, S6, E3, C3ne, M3ne, G5, P2v, Ai1, C6, M6, and D6 ECSs support VPC flow logs.

  For details about ECS types, see **ECS Types**.
- Each account can have up to 10 VPC flow logs in a region.

12.2 Creating a VPC Flow Log

Scenarios

A VPC flow log records information about the traffic going to and from a VPC.

Prerequisites

Ensure that the following operations have been performed on the LTS console:

- Create a log group.
- Create a log stream.

For more information about the LTS service, see the *Log Tank Service User Guide*.

Procedure

1. Go to the **VPC flow log list page**.
2. In the upper right corner, click **Create Flow Log**. On the displayed page, configure parameters as prompted.

Table 12-2 Parameter descriptions

Parameter	Description	Example Value
Name	The VPC flow log name. The name: • Can contain 1 to 64 characters. • Can contain letters, digits, underscores (_), hyphens (-), and periods (.).	flowlog-495d
Resource Type	Type of the resource whose traffic is to be logged. This parameter can only be set to **NIC**. • NIC • Subnet • VPC	NIC
Resource	The specific network interface whose traffic is to be logged. **NOTE** We recommend that you select an ECS in the running state. If an ECS in the stopped state is selected, restart the ECS after creating the VPC flow log for accurately recording the information about the traffic going to and from the ECS's network interface.	N/A
Filter	• **All traffic**: specifies that both accepted and rejected traffic of the specified resource will be logged. • **Accepted traffic**: specifies that only accepted traffic of the specified resource will be logged. Accepted traffic refers to the traffic permitted by the security group or network ACL. • **Rejected traffic**: specifies that only rejected traffic of the specified resource will be logged. Rejected traffic refers to the traffic denied by the network ACL.	All
Log Group	The log group created in LTS.	lts-group-abc
Log Stream	The log stream created in LTS.	lts-topic-abc

Parameter	Description	Example Value
Description	Supplementary information about the VPC flow log. This parameter is optional. The VPC flow log description can contain a maximum of 255 characters and cannot contain angle brackets (< or >).	N/A

NOTE

Only two flow logs, each with a different filter, can be created for a single resource under the same log group and log stream. Each VPC flow log must be unique.

3. Click **Create Now**.

   Return to the VPC flow log list. You can check the new VPC flow log.

12.3 Viewing a VPC Flow Log

Scenarios

This section describes how you can view the VPC flow log details.

The capture window is approximately 10 minutes, which indicates that a flow log record will be generated every 10 minutes. After creating a VPC flow log, you need to wait about 10 minutes before you can view the flow log record.

NOTE

If flow logs cannot be collected after the VPC flow log is enabled, the possible causes are as follows:

- If an ECS is in the stopped state, its flow log records will not be displayed.
- The VPC flow log quota is insufficient. If you want to continue collecting flow logs, **configure the quota**.

Procedure

1. Go to the **VPC flow log list page**.
2. Locate the target VPC flow log and click **View Log Record** in the **Operation** column to view information about the flow log record in LTS.

   The flow log record is in the following format:

   ```
   <version> <project-id> <interface-id> <srcaddr> <dstaddr> <srcport> <dstport> <protocol> <packets> <bytes> <start> <end> <action> <log-status>
   ```

   Table 12-3 provides you with flow log examples.

Table 12-3 Flow log examples

Scenario	Example Value
A flow log record in which data was recorded during the capture window	Value **1** indicates the VPC flow log version. Traffic with a size of 96 bytes to the network interface (**1d515d18-1b36-47dc-a983-bd6512aed4bd**) during the past 10 minutes (from 16:55:36 to 17:05:36 on January 29, 2019) was allowed. A data packet was transmitted over the UDP protocol from source IP address **192.168.0.154** and port **38929** to destination IP address **192.168.3.25** and port **53**. 1 5f67944957444bd6bb4fe3b367de8f3d 1d515d18-1b36-47dc-a983-bd6512aed4bd 192.168.0.154 192.168.3.25 38929 53 17 1 96 1548752136 1548752736 ACCEPT OK
A flow log record in which no data was recorded during the capture window	1 5f67944957444bd6bb4fe3b367de8f3d 1d515d18-1b36-47dc-a983-bd6512aed4bd - - - - - - - 1431280876 1431280934 - NODATA
A flow log record in which data was skipped during the capture window	1 5f67944957444bd6bb4fe3b367de8f3d 1d515d18-1b36-47dc-a983-bd6512aed4bd - - - - - - - 1431280876 1431280934 - SKIPDATA

You can enter a keyword on the log stream details page on the LTS console to search for flow log records.

12.4 Enabling or Disabling VPC Flow Log

Scenarios

After a VPC flow log is created, the VPC flow log is automatically enabled. If you do not need to record flow log data, you can disable the corresponding VPC flow log. A disabled VPC flow log can be enabled again.

Notes and Constraints

- After a VPC flow log is enabled, the system starts to collect flow logs in the next log collection period.
- After a VPC flow log is disabled, the system stops collecting flow logs in the next log collection period. Generated flow logs will still be reported.

Procedure

1. Go to the **VPC flow log list page**.
2. Locate the target flow log and click **Enable** or **Disable** in the **Operation** column.

   A confirmation dialog box is displayed.
3. Confirm the information and click **OK**.

12.5 Deleting a VPC Flow Log

Scenarios

You can delete a VPC flow log if you no longer need it. Deleting a VPC flow log will not delete the existing flow log records in LTS.

NOTE

If a network interface that uses a VPC flow log is deleted, the flow log will be automatically deleted. However, the flow log records are not deleted.

Procedure

1. Go to the **VPC flow log list page**.
2. Locate the target flow log and click **Delete** in the **Operation** column.

   A confirmation dialog box is displayed.
3. Confirm the information and click **OK**.

13 Traffic Mirroring

13.1 Traffic Mirroring

What Is Traffic Mirroring?

Traffic Mirroring can be used to mirror traffic that meets a mirror filter from mirror sources, such as elastic network interfaces. You can configure inbound and outbound rules for a mirror filter to determine which traffic will be mirrored from mirror sources to a mirror target, such as a network interface or load balancer. You can then send the traffic for inspection, audit analysis, and troubleshooting.

NOTICE

Currently, the Traffic Mirroring function is free. You will be notified in advance if the billing starts.

Currently, Traffic Mirroring is available only in certain regions. For details, visit **Function Overview** and click **Traffic Mirroring**.

Concepts

The following are the concepts for Traffic Mirroring:

- A mirror filter is a set of inbound rules and outbound rules to determine the traffic that is mirrored. You can specify matching criteria, such as priority and action for each rule.
  - Inbound rules match the traffic received by a mirror source.
  - Outbound rules match the traffic sent by a mirror source.
- A mirror source is an elastic network interface, from which traffic will be mirrored.
- A mirror target is a network interface of a cloud server or a load balancer, which is used to receive mirrored traffic.
- A mirror session can be associated with a mirror filter, multiple mirror sources, and a mirror target. A mirror session mirrors traffic from a mirror source to a mirror target that meets the mirror filter.

Working Principles

The following describes the working principles of traffic mirroring. As shown in **Figure 13-1**, a mirror session is associated with two mirror sources, one mirror filter, and one mirror target.

- Mirror source 01 is network interface-B that is attached to ECS-B. To access ECS-A from ECS-B, the outbound and inbound traffic of network interface-B is mirrored.
- Mirror source 02 is network interface-C that is attached to ECS-C. To access ECS-C from the Internet, the outbound and inbound traffic of network interface-C is mirrored.
- The mirror filter contains both inbound and outbound rules.
- The mirror target is a load balancer that receives mirrored traffic.

In **Table 13-1**, mirror sources network interface-B and network interface-C are used as examples to describe the traffic mirroring principle.

Figure 13-1 Traffic mirroring architecture

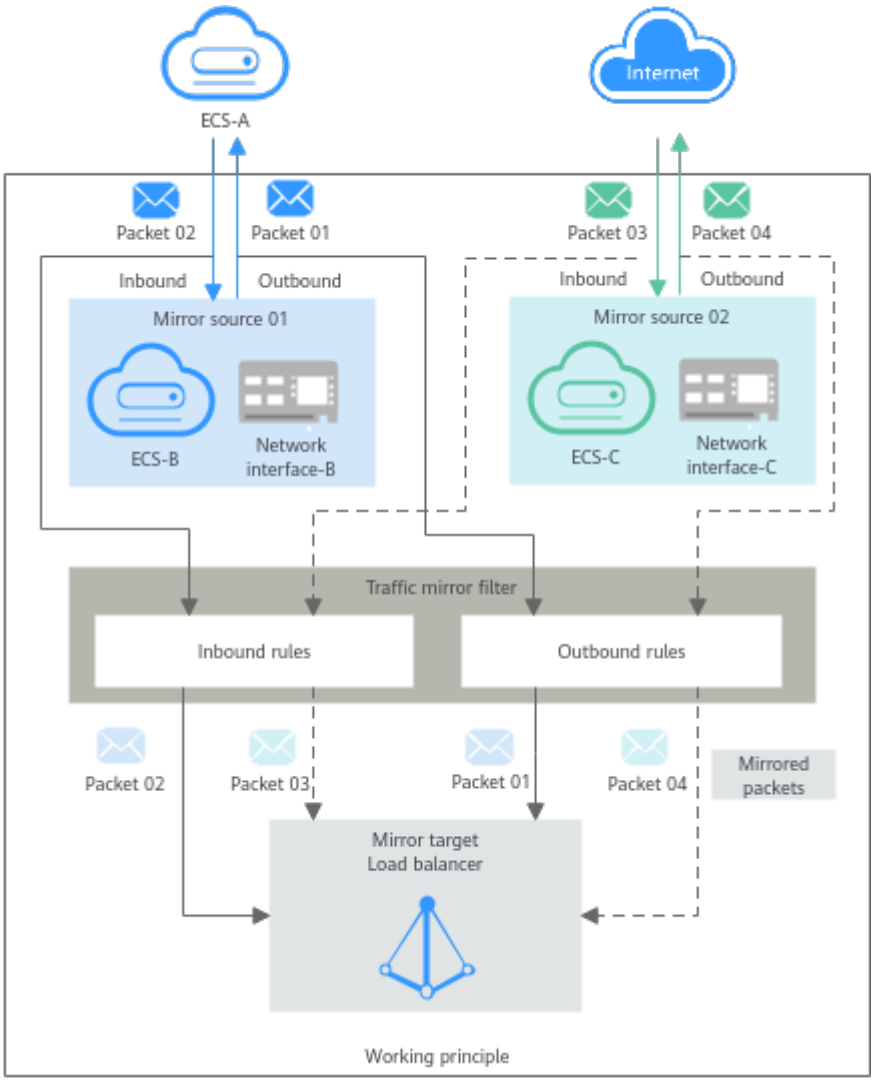


Table 13-1 Mirror path of packets

Mirror Source	Access Path	Packet	Direction	Description
Network interface-B	From ECS-B to ECS-A	Request packet 01	Outbound	Request packet 01 from ECS-B is an outbound packet for network interface-B. If packet 01 matches the outbound rules of the mirror filter, packet 01 is mirrored to the load balancer.
		Response packet 02	Inbound	Response packet 02 from ECS-A is an inbound packet for network interface-B. If packet 02 matches the inbound rules of the mirror filter, packet 02 is mirrored to the load balancer.
Network interface-C	From the Internet to ECS-C	Request packet 03	Inbound	Request packet 03 from the Internet is an inbound packet for network interface-C. If packet 03 matches the inbound rules of the mirror filter, packet 03 is mirrored to the load balancer.
		Response packet 04	Outbound	Response packet 04 from ECS-C is an outbound packet for network interface-C. If packet 04 matches the outbound rules of the mirror filter, packet 04 is mirrored to the load balancer.

Table 13-2 shows rules in a mirror filter and describes how the mirror session filters traffic using the mirror filter.

Table 13-2 Traffic filtering description

Direction	Priority	Protocol	Action	Type	Source	Source Port Range	Destination	Destination Port Range	Filtering Description
Inbound	1	TCP	Accept	IPv4	172.16.0.0/24	10000-10001	10.0.0.3/32	80-80	If traffic enters a network interface of the mirror source, the mirror session will mirror packets that meet the following rule: TCP (IPv4) packets from source 172.16.0.0/24 over port 10000 or 10001 to destination 10.0.0.3/32 over port 80
Outbound	1	All	Reject	IPv4	192.168.0.0/24	All	10.2.0.0/24	All	If traffic leaves a network interface of the mirror source, the mirror session will not mirror packets that meet the following rule: IPv4 packets from source 192.168.0.0/24 over any port to destination 10.2.0.0/24 over any port.

Application Scenarios

- Traffic inspection

  If there are network intrusions, you can use traffic mirroring to mirror required traffic to security software for comprehensively analysis and check. This helps to quickly locate security vulnerabilities and ensure network security.

- Traffic auditing

  You can use traffic mirroring to mirror traffic to a specific platform for auditing and analysis. This applies to scenarios that have high security requirements, such as finance.

- Fault locating

  O&M engineers can directly view mirrored traffic instead of capturing packets on service servers to locate faults. This prevents services from being affected during O&M.

Matching Rules

If a packet from the same mirror source meets multiple mirror filter rules, the packet is matched only once. The matching rules are described as follows:

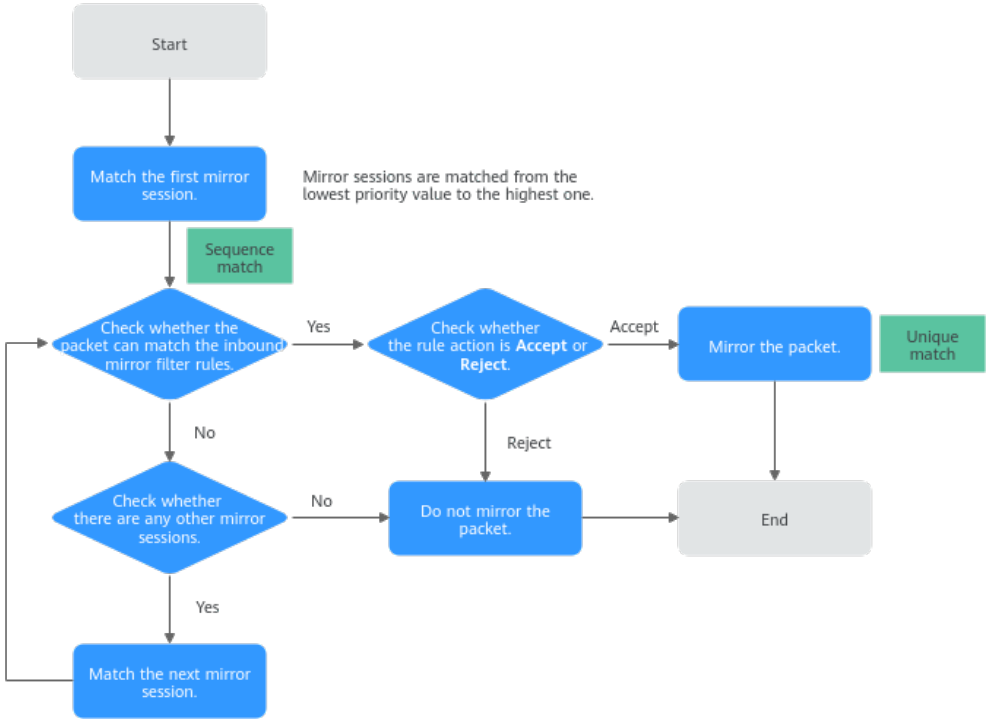
Table 13-3 Matching rules

Matching Rule	Description
Sequence match	Matching is performed in descending order of priority. A smaller value indicates a higher priority. For example, the priority of 1 is higher than that of 2. • Mirror session priority: A mirror source can be associated with multiple mirror sessions at the same time. The mirror sessions are matched in descending order of priority. For details, see **the matching process of mirror sessions**. • Mirror filter rule priority: A mirror session can be associated with only one mirror filter that contains multiple rules. The rules are matched in descending order of priority. An inbound or outbound mirror filter rule determines the traffic that is mirrored. You can specify matching criteria, such as priority and action for each rule. For details, see **the matching process of mirror filter rules**.
Unique match	If a packet matches a mirror filter rule, the packet does not attempt to match any other rules.

- **Figure 13-2** describes the matching process of mirror sessions. If a mirror source is associated with multiple mirror sessions, packets are matched in descending order of mirror session priorities. Inbound packets are used as an example here.
  - If a packet matches an inbound mirror filter rule of a mirror session:
    - The packet will be mirrored if the rule action is **Accept**.
    - The packet will not be mirrored if the rule action is **Reject**.
  - If a packet does not match any inbound mirror filter rule in a mirror session, the packet will not be mirrored.

  For example, a mirror source is associated with both mirror sessions A and B. The priority of mirror session A is 1 and that of mirror session B is 2. If a packet in the inbound direction of the mirror source meets the mirror filter rules of both mirror session A and mirror session B, the packet preferentially matches the mirror filter rules of mirror session A according to the priority, and will not match that of mirror session B.

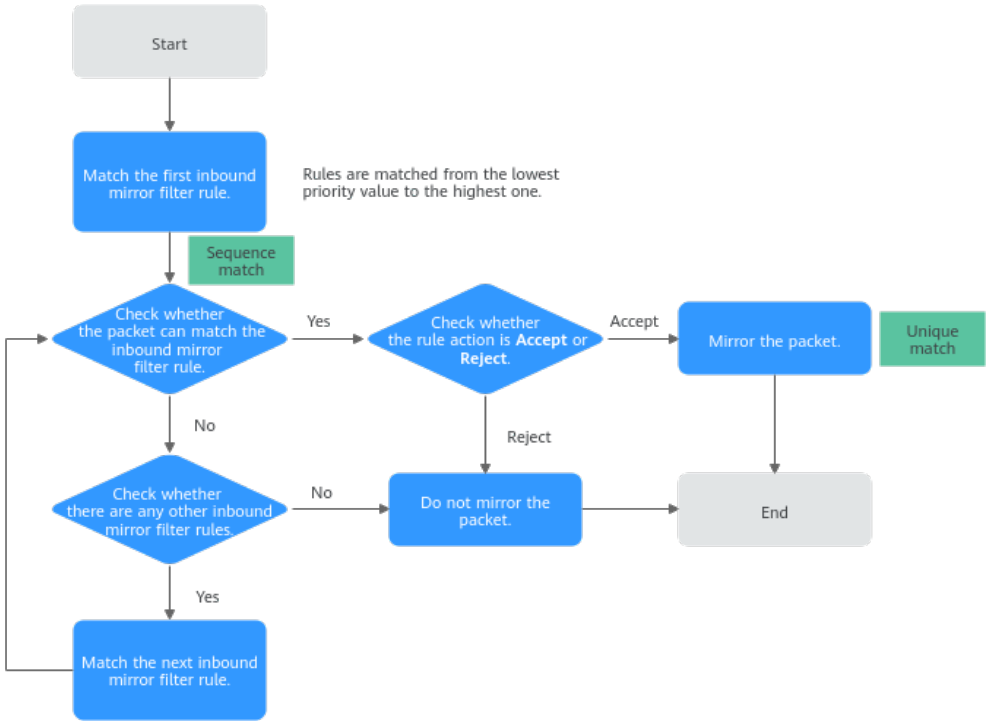
Figure 13-2 Mirror session matching process



- **Figure 13-3** describes the matching process of mirror filter rules. If a mirror source is associated with only one mirror session, packets are matched in descending order of priorities of inbound mirror filter rules. Inbound packets are used as an example here.
  - If a packet matches an inbound mirror filter rule:
    - The packet will be mirrored if the rule action is **Accept**.
    - The packet will not be mirrored if the rule action is **Reject**.
  - If a packet does not match any inbound mirror filter rule, the packet will not be mirrored.

  For example, a mirror source is associated with mirror session A. The mirror filter of mirror session A has inbound rules A and B, which have the same traffic matching conditions but different priorities and actions. The priority of rule A is 1, and the action is **Reject**. The priority of rule B is 2, and the action is **Accept**. If a packet in the inbound direction of the mirror source meets the traffic matching conditions of both rule A and rule B, the packet matches rule A first according to the rule priority. The packet will be rejected and will not be mirrored and match rule B.

Figure 13-3 Mirror filter rule matching process



Traffic Mirroring Quotas

Table 13-4 lists the quotas about Traffic Mirroring resources. Some default quotas can be increased.

Table 13-4 Quotas

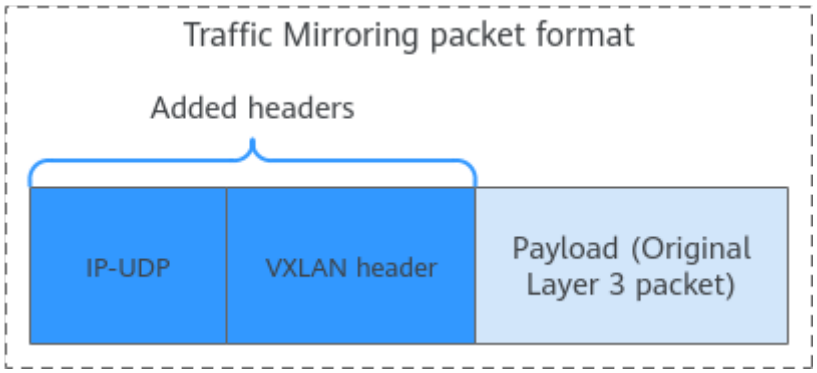
Item	Default Quota	Adjustable
Maximum number of mirror sources that can be associated with a mirror session	10	Yes. For details, see Managing Quotas.
Maximum number of mirror sessions that can be associated with a mirror source	3	No
Maximum number of mirror targets that can be associated with a mirror session	1	No

Item	Default Quota	Adjustable
Maximum number of mirror sessions that can be associated with a mirror target	• 10 (if the mirror target is the network interface of a cloud server) • 200 (if the mirror target is a load balancer)	No
Maximum number of mirror filters that can be associated with a mirror session	1	No
Maximum number of mirror sessions that can be associated with a mirror filter	1,000	No
Maximum number of rules that can be added to a mirror filter	• 10 inbound rules • 10 outbound rules	No
Maximum number of mirror sessions that can be created in a region	20,000	No

Notes and Constraints

- As shown in **Figure 13-4**, mirrored traffic is encapsulated in the standard VXLAN packet format. If the total length of mirrored packets and VXLAN packets is greater than the MTU of the mirror source, the system truncates the packets. To prevent packets from being truncated, you are advised to set the MTU of the elastic network interface to be at least 64 bytes smaller than the MTU supported by the link in IPv4 scenarios.

  Figure 13-4 Traffic Mirroring packet format

- **Table 13-5** and **Table 13-6** show the constraints on different types of mirror sources and mirror targets.

Table 13-5 Constraints on mirror sources

Mirror Source Type	Constraints
Elastic network interface	• If the mirror source is an elastic network interface, the network interface needs to be attached to an ECS. Only the network interface of an ECS with certain flavors (such as C7t and aC7) can be used as mirror sources. You can call APIs to **query details about ECS flavors** and use the response value of **network_interface:traffic_mirroring_supported** to check whether an ECS flavor supports traffic mirroring. • An elastic network interface cannot be used as both a mirror source and a mirror target at the same time. • Traffic Mirroring occupies the bandwidth of instances attached to elastic network interfaces and does not have bandwidth limits.

Table 13-6 Constraints on mirror targets

Mirror Target Type	Constraints
Network interface	• If a mirror target needs to receive mirrored traffic from multiple mirror sources, ensure that the mirror target has proper specifications based on service requirements. • An elastic network interface cannot be used as both a mirror source and a mirror target at the same time.
Load balancer	The encapsulated mirrored packet uses the IPv4 UDP protocol. So the dedicated load balancer used as the mirror target must support IPv4 UDP.

- If a packet from a mirror source meets multiple mirror filter rules, the packet will be matched only once and will be accepted or rejected to a mirror target according to the rule action.
- If a packet from a mirror source is discarded by a security group or network ACL, the packet will not be mirrored.
- If a packet from a mirror source meets a mirror filter, the packet will be mirrored and will not be restricted by outbound rules of a security group or network ACL of the mirror source. This means you do not need to configure the security group or network ACL for the mirror source. However, if you want to mirror the packet to the mirror target, you need to configure the following rules for the security group and network ACL of the mirror target:
  - Add a security group rule to allow inbound UDP packets from the mirror source over port 4789.

Table 13-7 shows a rule example if the private IP address of the mirror source is 192.168.0.27. To learn about how to add a rule, see **Adding a Security Group Rule**.

Table 13-7 Security group rule configuration example (a network interface as the mirror source)

Direction	Action	Type	Protocol & Port	Source
Inbound	Allow	IPv4	UDP: 4789	192.168.0.27/32 Set the source based on the actual requirements.

- Add a network ACL rule to allow inbound UDP packets from the mirror source over port 4789.

  Table 13-8 shows a rule example if the IP address of the mirror source is 192.168.0.27. To learn about how to add a rule, see **Adding a Network ACL Rule**.

Table 13-8 Network ACL rule configuration example (a network interface as the mirror source)

Direction	Type	Action	Protocol	Source	Source Port Range	Destination	Destination Port Range
Inbound	IPv4	Allow	UDP	192.168.0.27/32 Set the source based on the actual requirements.	If not specified, all ports are used.	• If the mirror target is a network interface, configure its private IPv4 address as the destination, for example, 192.168.1.24/32. • If mirror target is a load balancer, configure its private IPv4 address as the destination, for example, 192.168.1.25/32. Set the destination based on the actual requirements. Ensure that the IP	4789 Port 4789 must be opened. Open other ports based on the actual requirements.

Direction	Type	Action	Protocol	Source	Source Port Range	Destination	Destination Port Range
						address of the mirror target is within the destination CIDR block.	

- Resources from different VPCs cannot communicate with each other. If a mirror source and a mirror target are not in the same VPC, you need to use a VPC peering connection or an enterprise router to connect their VPCs first.
  - To use a VPC peering connection, see **VPC Peering Connection Overview**.
  - To use an enterprise router, see **Using an Enterprise Router to Enable Communications Between VPCs in the Same Region**.

Usage Process

To use the traffic mirroring function, you need to create a mirror session and associate a mirror filter, mirror sources, and a mirror target with the mirror session. **Figure 13-5** shows the process.

Figure 13-5 Process of using Traffic Mirroring

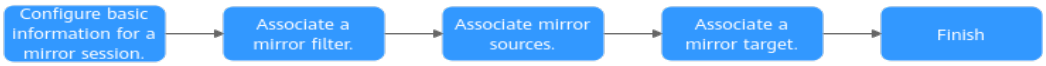


Table 13-9 Description of the Traffic Mirroring process

Step	Description	Reference
Configure basic information about a mirror session.	Set parameters such as the name and priority of the mirror session.	**Creating a Mirror Session**
Associate a mirror filter.	Select a mirror filter and associate it with the mirror session. Each mirror session can have one mirror filter associated. If there is no mirror filter required, you can create one by referring to **Creating a Mirror Filter**.	

Step	Description	Reference
Associate mirror sources.	Select an elastic network interface as the mirror source and associate it with the mirror session. Each mirror session can be associated with multiple mirror sources.	
Associate a mirror target.	Select the network interface of a cloud server or load balancer as the mirror target and associate it with the mirror session.	
Finish	If the mirror session is enabled, the traffic that meets the mirror filter from the mirror source will be mirrored to the mirror target. If you do not enable the mirror session when creating it, the traffic of the mirror source will not be mirrored. You can enable the mirror session by referring to **Enabling or Disabling a Mirror Session**.	

13.2 Mirror Filters

13.2.1 Creating a Mirror Filter

Scenarios

A mirror filter is a set of inbound rules and outbound rules to determine the traffic that is mirrored. You can specify matching criteria, such as priority and action for each rule.

- Inbound rules match the traffic received by a mirror source.
- Outbound rules match the traffic sent by a mirror source.

A mirror filter takes effect only after it is associated with mirror sessions.

Mirror Filter Rule Examples

Table 13-10 shows rules in a mirror filter and describes how the mirror session filters traffic using the mirror filter.

Table 13-10 Traffic filtering description

Direction	Priority	Protocol	Action	Type	Source	Source Port Range	Destination	Destination Port Range	Filtering Description
Inbound	1	TCP	Accept	IPv4	172.16.0.0/24	10000-10001	10.0.0.3/32	80-80	If traffic enters a network interface of the mirror source, the mirror session will mirror packets that meet the following rule: TCP (IPv4) packets from source 172.16.0.0/24 over port 10000 or 10001 to destination 10.0.0.3/32 over port 80
Outbound	1	All	Reject	IPv4	192.168.0.0/24	All	10.2.0.0/24	All	If traffic leaves a network interface of the mirror source, the mirror session will not mirror packets that meet the following rule: IPv4 packets from source 192.168.0.0/24 over any port to destination 10.2.0.0/24 over any port.

Procedure

1. Go to the **mirror filter list page**.
2. In the upper right corner of the mirror filter list, click **Create Mirror Filter**.
   The **Create Mirror Filter** page is displayed.
3. Set basic information about the mirror filter as prompted.

   Table 13-11 Parameters for configuring basic information

Parameter	Description	Example Value
Name	Mandatory Enter the name of the mirror filter. The name: • Must contain 1 to 64 characters. • Can contain letters, digits, underscores (_), hyphens (-), and periods (.).	mirror-filter-01

Parameter	Description	Example Value
Description	Optional Enter the description of the mirror filter in the text box as required.	-

4. Click **Add Rule** in the **Inbound Rules** area to add inbound rules.

   You can click ⊕ to add more inbound rules.

Table 13-12 Inbound rule parameter description

Parameter	Description	Example Value
Priority	Priority of a mirror filter rule. • A priority value can be from 1 to 65535. A smaller value indicates a higher priority. • Priorities of inbound rules must be unique for each mirror filter. A mirror filter can contain multiple rules and the rules are matched in ascending order of priority. For details, see **the matching process of mirror filter rules**.	1
Protocol	Select a network protocol. • If you select TCP, you can customize the source and destination port ranges. • If you select UDP, you can customize the source and destination port ranges. • If you set **Type** to **IPv4** and select ICMP, all ports are specified for source and destination port ranges by default. • If you set **Type** to **IPv6** and select ICMPv6, all ports are specified for source and destination port ranges by default. • If you select **All**, all network protocols are supported and all ports are specified for source and destination port ranges by default.	TCP
Action	Whether to accept or reject inbound traffic of a mirror source. • If you set **Action** to **Accept**, the traffic will be mirrored to the mirror target. • If you set **Action** to **Reject**, the traffic will not be mirrored to the mirror target.	Accept

Parameter	Description	Example Value
Type	IP address version of inbound traffic. You can specify: • **IPv4** • **IPv6**	IPv4
Source	Source of inbound traffic. You can enter: • A single IP address: IP address/mask Example IPv4 address: 192.168.10.10/32 Example IPv6 address: 2002:50::44/128 • An IP address range in CIDR notation: IP address/mask Example IPv4 address range: 192.168.52.0/24 Example IPv6 address range: 2407:c080:802:469::/64 • All IP addresses 0.0.0.0/0 represents all IPv4 addresses. ::/0 represents all IPv6 addresses.	10.0.0.0/24
Source Port Range	Source port range of inbound traffic. • Port range: 1 to 65535 • Use a hyphen (-) to connect the start port and the end port, for example, 22-23. The end port cannot be smaller than the start port. • If not specified or **1-65535** is specified, all ports are used.	22-23
Destination	Destination of inbound traffic. You can enter: • A single IP address: IP address/mask Example IPv4 address: 192.168.10.10/32 Example IPv6 address: 2002:50::44/128 • An IP address range in CIDR notation: IP address/mask Example IPv4 address range: 192.168.52.0/24 Example IPv6 address range: 2407:c080:802:469::/64 • All IP addresses 0.0.0.0/0 represents all IPv4 addresses. ::/0 represents all IPv6 addresses.	0.0.0.0/0

Parameter	Description	Example Value
Destination Port Range	Destination port range of inbound traffic. • Port range: 1 to 65535 • Use a hyphen (-) to connect the start port and the end port, for example, 22-23. The end port cannot be smaller than the start port. • If not specified or **1-65535** is specified, all ports are used.	1-65535
Description	Enter the description of the mirror filter rule in the text box as required.	-

5. Click **OK**.
6. Click **Add Rule** in the **Outbound Rules** area to add outbound rules.

   You can click ⊕ to add more outbound rules.

Table 13-13 Outbound rule parameter description

Parameter	Description	Example Value
Priority	Priority of a mirror filter rule. • A priority value can be from 1 to 65535. A smaller value indicates a higher priority. • Priorities of inbound rules must be unique for each mirror filter. A mirror filter can contain multiple rules and the rules are matched in ascending order of priority. For details, see **the matching process of mirror filter rules**.	1
Protocol	Select a network protocol. • If you select TCP, you can customize the source and destination port ranges. • If you select UDP, you can customize the source and destination port ranges. • If you set **Type** to **IPv4** and select ICMP, all ports are specified for source and destination port ranges by default. • If you set **Type** to **IPv6** and select ICMPv6, all ports are specified for source and destination port ranges by default. • If you select **All**, all network protocols are supported and all ports are specified for source and destination port ranges by default.	All

Parameter	Description	Example Value
Action	Whether to accept or reject outbound traffic of a mirror source. • If you set **Action** to **Accept**, the traffic will be mirrored to the mirror target. • If you set **Action** to **Reject**, the traffic will not be mirrored to the mirror target.	Reject
Type	IP address version of outbound traffic. You can specify: • **IPv4** • **IPv6**	IPv4
Source	Source of outbound traffic. You can enter: • A single IP address: IP address/mask Example IPv4 address: 192.168.10.10/32 Example IPv6 address: 2002:50::44/128 • An IP address range in CIDR notation: IP address/mask Example IPv4 address range: 192.168.52.0/24 Example IPv6 address range: 2407:c080:802:469::/64 • All IP addresses 0.0.0.0/0 represents all IPv4 addresses. ::/0 represents all IPv6 addresses.	192.168.0.0/24
Source Port Range	Source port range of outbound traffic. • Port range: 1 to 65535 • Use a hyphen (-) to connect the start port and the end port, for example, 22-23. The end port cannot be smaller than the start port. • If not specified or **1-65535** is specified, all ports are used.	All
Destination	Destination of outbound traffic. You can enter: • A single IP address: IP address/mask Example IPv4 address: 192.168.10.10/32 Example IPv6 address: 2002:50::44/128 • An IP address range in CIDR notation: IP address/mask Example IPv4 address range: 192.168.52.0/24 Example IPv6 address range: 2407:c080:802:469::/64 • All IP addresses 0.0.0.0/0 represents all IPv4 addresses. ::/0 represents all IPv6 addresses.	10.2.0.0/24

Parameter	Description	Example Value
Destination Port Range	Destination port range of outbound traffic. • Port range: 1 to 65535 • Use a hyphen (-) to connect the start port and the end port, for example, 22-23. The end port cannot be smaller than the start port. • If not specified or **1-65535** is specified, all ports are used.	All
Description	Enter the description of the mirror filter rule in the text box as required.	-

7. Click **OK**.
8. After setting the parameters, click **Create Now**.
   The mirror filter list page is displayed.

Follow-up Operations

A mirror filter takes effect only after it is associated with mirror sessions. Each mirror session only can have one mirror filter associated.

- If you have no mirror session, refer to **Creating a Mirror Session**.
- If you have a mirror session and want to change the mirror filter of the mirror session, refer to **Changing the Mirror Filter for a Mirror Session**.

13.2.2 Adding an Inbound or Outbound Mirror Filter Rule

Scenarios

You can add inbound and outbound rules to a mirror filter.

- Inbound rules match the traffic received by a mirror source.
- Outbound rules match the traffic sent by a mirror source.

Mirror Filter Rule Examples

Table 13-14 shows rules in a mirror filter and describes how the mirror session filters traffic using the mirror filter.

Table 13-14 Traffic filtering description

Direction	Priority	Protocol	Action	Type	Source	Source Port Range	Destination	Destination Port Range	Filtering Description
Inbound	1	TCP	Accept	IPv4	172.16.0.0/24	10000-10001	10.0.0.3/32	80-80	If traffic enters a network interface of the mirror source, the mirror session will mirror packets that meet the following rule: TCP (IPv4) packets from source 172.16.0.0/24 over port 10000 or 10001 to destination 10.0.0.3/32 over port 80
Outbound	1	All	Reject	IPv4	192.168.0.0/24	All	10.2.0.0/24	All	If traffic leaves a network interface of the mirror source, the mirror session will not mirror packets that meet the following rule: IPv4 packets from source 192.168.0.0/24 over any port to destination 10.2.0.0/24 over any port.

Procedure

1. Go to the **mirror filter list page**.
2. Locate the target mirror filter and click the number in the **Inbound and Outbound Rules** column.

   The **Inbound Rules** tab page is displayed.
3. On the **Inbound Rules** tab, click **Add Rule**. In the displayed dialog box, set required parameters to add an inbound rule.

   You can click ⊕ to add more inbound rules.

Table 13-15 Inbound rule parameter description

Parameter	Description	Example Value
Priority	Priority of a mirror filter rule. • A priority value can be from 1 to 65535. A smaller value indicates a higher priority. • Priorities of inbound rules must be unique for each mirror filter. A mirror filter can contain multiple rules and the rules are matched in ascending order of priority. For details, see **the matching process of mirror filter rules**.	1
Protocol	Select a network protocol. • If you select TCP, you can customize the source and destination port ranges. • If you select UDP, you can customize the source and destination port ranges. • If you set **Type** to **IPv4** and select ICMP, all ports are specified for source and destination port ranges by default. • If you set **Type** to **IPv6** and select ICMPv6, all ports are specified for source and destination port ranges by default. • If you select **All**, all network protocols are supported and all ports are specified for source and destination port ranges by default.	TCP
Action	Whether to accept or reject inbound traffic of a mirror source. • If you set **Action** to **Accept**, the traffic will be mirrored to the mirror target. • If you set **Action** to **Reject**, the traffic will not be mirrored to the mirror target.	Accept
Type	IP address version of inbound traffic. You can specify: • **IPv4** • **IPv6**	IPv4

Parameter	Description	Example Value
Source	Source of inbound traffic. You can enter: • A single IP address: IP address/mask Example IPv4 address: 192.168.10.10/32 Example IPv6 address: 2002:50::44/128 • An IP address range in CIDR notation: IP address/mask Example IPv4 address range: 192.168.52.0/24 Example IPv6 address range: 2407:c080:802:469::/64 • All IP addresses 0.0.0.0/0 represents all IPv4 addresses. ::/0 represents all IPv6 addresses.	10.0.0.0/24
Source Port Range	Source port range of inbound traffic. • Port range: 1 to 65535 • Use a hyphen (-) to connect the start port and the end port, for example, 22-23. The end port cannot be smaller than the start port. • If not specified or **1-65535** is specified, all ports are used.	22-23
Destination	Destination of inbound traffic. You can enter: • A single IP address: IP address/mask Example IPv4 address: 192.168.10.10/32 Example IPv6 address: 2002:50::44/128 • An IP address range in CIDR notation: IP address/mask Example IPv4 address range: 192.168.52.0/24 Example IPv6 address range: 2407:c080:802:469::/64 • All IP addresses 0.0.0.0/0 represents all IPv4 addresses. ::/0 represents all IPv6 addresses.	0.0.0.0/0
Destination Port Range	Destination port range of inbound traffic. • Port range: 1 to 65535 • Use a hyphen (-) to connect the start port and the end port, for example, 22-23. The end port cannot be smaller than the start port. • If not specified or **1-65535** is specified, all ports are used.	1-65535
Description	Enter the description of the mirror filter rule in the text box as required.	-

4. Click **OK**.

   You can view the added inbound rule in the list.

5. Click the **Outbound Rules** tab. In the upper left corner of the outbound rule list, click **Add Rule** to add an outbound rule.

   You can click ⊕ to add more outbound rules.

Table 13-16 Outbound rule parameter description

Parameter	Description	Example Value
Priority	Priority of a mirror filter rule. • A priority value can be from 1 to 65535. A smaller value indicates a higher priority. • Priorities of inbound rules must be unique for each mirror filter. A mirror filter can contain multiple rules and the rules are matched in ascending order of priority. For details, see **the matching process of mirror filter rules**.	1
Protocol	Select a network protocol. • If you select TCP, you can customize the source and destination port ranges. • If you select UDP, you can customize the source and destination port ranges. • If you set **Type** to **IPv4** and select ICMP, all ports are specified for source and destination port ranges by default. • If you set **Type** to **IPv6** and select ICMPv6, all ports are specified for source and destination port ranges by default. • If you select **All**, all network protocols are supported and all ports are specified for source and destination port ranges by default.	All
Action	Whether to accept or reject outbound traffic of a mirror source. • If you set **Action** to **Accept**, the traffic will be mirrored to the mirror target. • If you set **Action** to **Reject**, the traffic will not be mirrored to the mirror target.	Reject
Type	IP address version of outbound traffic. You can specify: • **IPv4** • **IPv6**	IPv4

Parameter	Description	Example Value
Source	Source of outbound traffic. You can enter: • A single IP address: IP address/mask Example IPv4 address: 192.168.10.10/32 Example IPv6 address: 2002:50::44/128 • An IP address range in CIDR notation: IP address/mask Example IPv4 address range: 192.168.52.0/24 Example IPv6 address range: 2407:c080:802:469::/64 • All IP addresses 0.0.0.0/0 represents all IPv4 addresses. ::/0 represents all IPv6 addresses.	192.168.0.0/24
Source Port Range	Source port range of outbound traffic. • Port range: 1 to 65535 • Use a hyphen (-) to connect the start port and the end port, for example, 22-23. The end port cannot be smaller than the start port. • If not specified or **1-65535** is specified, all ports are used.	All
Destination	Destination of outbound traffic. You can enter: • A single IP address: IP address/mask Example IPv4 address: 192.168.10.10/32 Example IPv6 address: 2002:50::44/128 • An IP address range in CIDR notation: IP address/mask Example IPv4 address range: 192.168.52.0/24 Example IPv6 address range: 2407:c080:802:469::/64 • All IP addresses 0.0.0.0/0 represents all IPv4 addresses. ::/0 represents all IPv6 addresses.	10.2.0.0/24
Destination Port Range	Destination port range of outbound traffic. • Port range: 1 to 65535 • Use a hyphen (-) to connect the start port and the end port, for example, 22-23. The end port cannot be smaller than the start port. • If not specified or **1-65535** is specified, all ports are used.	All
Description	Enter the description of the mirror filter rule in the text box as required.	-

6. Click **OK**.
   You can view the added outbound rule in the list.

13.2.3 Modifying an Inbound or Outbound Mirror Filter Rule

Scenarios

You can modify inbound and outbound rules of a mirror filter.

- Inbound rules match the traffic received by a mirror source.
- Outbound rules match the traffic sent by a mirror source.

Procedure

1. Go to the **mirror filter list page**.
2. Locate the target mirror filter and click the number in the **Inbound and Outbound Rules** column.
   The **Inbound Rules** tab page is displayed.
3. In the inbound rule list, locate the target rule and click **Modify** in the **Operation** column.

Table 13-17 Inbound rule parameter description

Parameter	Description	Example Value
Priority	Priority of a mirror filter rule. • A priority value can be from 1 to 65535. A smaller value indicates a higher priority. • Priorities of inbound rules must be unique for each mirror filter. A mirror filter can contain multiple rules and the rules are matched in ascending order of priority. For details, see **the matching process of mirror filter rules**.	1

Parameter	Description	Example Value
Protocol	Select a network protocol. • If you select TCP, you can customize the source and destination port ranges. • If you select UDP, you can customize the source and destination port ranges. • If you set **Type** to **IPv4** and select ICMP, all ports are specified for source and destination port ranges by default. • If you set **Type** to **IPv6** and select ICMPv6, all ports are specified for source and destination port ranges by default. • If you select **All**, all network protocols are supported and all ports are specified for source and destination port ranges by default.	TCP
Action	Whether to accept or reject inbound traffic of a mirror source. • If you set **Action** to **Accept**, the traffic will be mirrored to the mirror target. • If you set **Action** to **Reject**, the traffic will not be mirrored to the mirror target.	Accept
Type	IP address version of inbound traffic. You can specify: • **IPv4** • **IPv6**	IPv4
Source	Source of inbound traffic. You can enter: • A single IP address: IP address/mask Example IPv4 address: 192.168.10.10/32 Example IPv6 address: 2002:50::44/128 • An IP address range in CIDR notation: IP address/mask Example IPv4 address range: 192.168.52.0/24 Example IPv6 address range: 2407:c080:802:469::/64 • All IP addresses 0.0.0.0/0 represents all IPv4 addresses. ::/0 represents all IPv6 addresses.	10.0.0.0/24

Parameter	Description	Example Value
Source Port Range	Source port range of inbound traffic. • Port range: 1 to 65535 • Use a hyphen (-) to connect the start port and the end port, for example, 22-23. The end port cannot be smaller than the start port. • If not specified or **1-65535** is specified, all ports are used.	22-23
Destination	Destination of inbound traffic. You can enter: • A single IP address: IP address/mask Example IPv4 address: 192.168.10.10/32 Example IPv6 address: 2002:50::44/128 • An IP address range in CIDR notation: IP address/mask Example IPv4 address range: 192.168.52.0/24 Example IPv6 address range: 2407:c080:802:469::/64 • All IP addresses 0.0.0.0/0 represents all IPv4 addresses. ::/0 represents all IPv6 addresses.	0.0.0.0/0
Destination Port Range	Destination port range of inbound traffic. • Port range: 1 to 65535 • Use a hyphen (-) to connect the start port and the end port, for example, 22-23. The end port cannot be smaller than the start port. • If not specified or **1-65535** is specified, all ports are used.	1-65535
Description	Enter the description of the mirror filter rule in the text box as required.	-

4. Click **OK**.

   You can view the modified inbound rule in the list.

5. On the **Outbound Rules** tab page, locate the row that contains the rule in the outbound rule list and click **Modify** in the **Operation** column.

Table 13-18 Outbound rule parameter description

Parameter	Description	Example Value
Priority	Priority of a mirror filter rule. • A priority value can be from 1 to 65535. A smaller value indicates a higher priority. • Priorities of inbound rules must be unique for each mirror filter. A mirror filter can contain multiple rules and the rules are matched in ascending order of priority. For details, see **the matching process of mirror filter rules**.	1
Protocol	Select a network protocol. • If you select TCP, you can customize the source and destination port ranges. • If you select UDP, you can customize the source and destination port ranges. • If you set **Type** to **IPv4** and select ICMP, all ports are specified for source and destination port ranges by default. • If you set **Type** to **IPv6** and select ICMPv6, all ports are specified for source and destination port ranges by default. • If you select **All**, all network protocols are supported and all ports are specified for source and destination port ranges by default.	All
Action	Whether to accept or reject outbound traffic of a mirror source. • If you set **Action** to **Accept**, the traffic will be mirrored to the mirror target. • If you set **Action** to **Reject**, the traffic will not be mirrored to the mirror target.	Reject
Type	IP address version of outbound traffic. You can specify: • **IPv4** • **IPv6**	IPv4

Parameter	Description	Example Value
Source	Source of outbound traffic. You can enter: • A single IP address: IP address/mask Example IPv4 address: 192.168.10.10/32 Example IPv6 address: 2002:50::44/128 • An IP address range in CIDR notation: IP address/mask Example IPv4 address range: 192.168.52.0/24 Example IPv6 address range: 2407:c080:802:469::/64 • All IP addresses 0.0.0.0/0 represents all IPv4 addresses. ::/0 represents all IPv6 addresses.	192.168.0.0/24
Source Port Range	Source port range of outbound traffic. • Port range: 1 to 65535 • Use a hyphen (-) to connect the start port and the end port, for example, 22-23. The end port cannot be smaller than the start port. • If not specified or **1-65535** is specified, all ports are used.	All
Destination	Destination of outbound traffic. You can enter: • A single IP address: IP address/mask Example IPv4 address: 192.168.10.10/32 Example IPv6 address: 2002:50::44/128 • An IP address range in CIDR notation: IP address/mask Example IPv4 address range: 192.168.52.0/24 Example IPv6 address range: 2407:c080:802:469::/64 • All IP addresses 0.0.0.0/0 represents all IPv4 addresses. ::/0 represents all IPv6 addresses.	10.2.0.0/24
Destination Port Range	Destination port range of outbound traffic. • Port range: 1 to 65535 • Use a hyphen (-) to connect the start port and the end port, for example, 22-23. The end port cannot be smaller than the start port. • If not specified or **1-65535** is specified, all ports are used.	All
Description	Enter the description of the mirror filter rule in the text box as required.	-

6. Click **OK**.

   You can view the modified outbound rule in the list.

13.2.4 Deleting an Inbound or Outbound Mirror Filter Rule

Scenarios

You can delete inbound and outbound rules of a mirror filter.

Procedure

1. Go to the **mirror filter list page**.
2. Locate the target mirror filter and click the number in the **Inbound and Outbound Rules** column.

   The **Inbound Rules** tab page is displayed.
3. Locate the target inbound rule and click **Delete** in the **Operation** column.

   A confirmation dialog box is displayed.
4. Click **OK**.

   Deleted inbound rules cannot be recovered.
5. On the **Outbound Rules** tab page, locate the row that contains the rule in the outbound rule list and click **Delete** in the **Operation** column.

   A confirmation dialog box is displayed.
6. Click **OK**.

   Deleted outbound rules cannot be recovered.

13.2.5 Modifying the Basic Information About a Mirror Filter

Scenarios

You can modify basic information about a mirror filter, including its name and description.

Procedure

1. Go to the **mirror filter list page**.
2. Locate the target mirror filter and click its name.

   The **Inbound Rules** tab is displayed.
3. Click the **Basic Information** tab and modify parameters as prompted.

   a. Click  next to the parameter to be modified and enter information in the text box.

   b. Click  to save the modification.

Table 13-19 Parameters that can be modified

Parameter	Description	Example Value
Name	Mandatory Enter a different name for the mirror filter. The name: • Can contain 1 to 64 characters. • Can contain letters, digits, underscores (_), hyphens (-), and periods (.).	mirror-filter-01
Description	Optional Enter a description for the mirror filter in the text box as required.	-

13.2.6 Viewing the Details About a Mirror Filter

Scenarios

You can view the following information about a mirror filter:

- Basic information, such as name, ID, and creation time
- Inbound and outbound rules, such as their priority, protocol, and action
- Associated mirror sessions, such as their name, mirror target, and status

Procedure

1. Go to the **mirror filter list page**.
2. Locate the target mirror filter and click its name.

   The **Inbound Rules** tab page is displayed.
3. View information about the mirror filter on different tab pages.
   - On the **Basic Information** tab page, view mirror filter information, such as name, ID, and creation time.
   - On the **Inbound Rules** tab page, view rule details, such as their priority, protocol, and action.
   - On the **Outbound Rules** tab page, view rule details, such as their priority, protocol, and action.
   - On the **Associated Mirror Sessions** tab page, view information about mirror sessions, such as their name, mirror target, and status.

13.2.7 Deleting a Mirror Filter

Scenarios

If a mirror filter is no longer required, you can delete it.

Notes and Constraints

If a mirror filter has mirror sessions associated, disassociate the mirror sessions first and then delete the mirror filter.

- Each mirror session must have a mirror filter associated. You can change the mirror filter for the mirror sessions. For details, see **Changing the Mirror Filter for a Mirror Session**.
- If your mirror sessions are no longer required, you can also delete them. For details, see **Deleting a Mirror Session**.

Procedure

1. Go to the **mirror filter list page**.
2. Locate the target mirror filter and click **Delete** in the **Operation** column.

   A confirmation dialog box is displayed.
3. Confirm the information and click **OK**.

   A deleted mirror filter cannot be recovered.

13.3 Mirror Sessions

13.3.1 Creating a Mirror Session

Scenarios

To use Traffic Mirroring, you need to create a mirror session, associate a mirror filter, multiple mirror sources, and a mirror target with the mirror session. A mirror session mirrors traffic from a mirror source to a mirror target that meets the mirror filter.

For details about mirror sessions, see **Traffic Mirroring**.

Procedure

1. Go to the **mirror session list page**.
2. In the upper right corner of the mirror session list, click **Create Mirror Session**.

   The **Create Mirror Session** page is displayed.
3. Set basic information about the mirror session as prompted.

Table 13-20 Parameters for configuring basic information about a mirror session

Parameter	Description	Example Value
Name	Mandatory Enter the name of the mirror session. The name: • Must contain 1 to 64 characters. • Can contain letters, digits, underscores (_), hyphens (-), and periods (.).	mirror-session-01
Priority	Mandatory Priority of the mirror session. • A priority value can be from 1 to 32766. A smaller value indicates a higher priority. • Priorities must be unique for each mirror session of the same account in a region. A mirror source can be associated with multiple mirror sessions at the same time. The mirror sessions are matched from the lowest value to the highest value. For details, see **the matching process of mirror sessions**.	1
VNI	Optional VXLAN Network Identifiers (VNIs) are used to distinguish different mirror sessions for a mirror target. A VNI can be from 0 to 16777215. If not specified, the default value is 1.	1
Packet Length	Optional The number of bytes that meet the mirror filter and will be mirrored. The value can be from 1 to 1460. If not specified, the default value is 96.	96
Mirror Session	Optional • If the mirror session is disabled, traffic of mirror sources cannot be monitored. • If the mirror session is enabled, traffic of mirror sources can be monitored.	Enable
Description	Optional Enter the description of the mirror session in the text box as required.	-

4. Click **Next**.

   The **Associate Mirror Filter** page is displayed.

5. In the mirror filter list, select a mirror filter.

   Each mirror session can be associated with only one mirror filter.

   If there is no mirror filter you want, create one by referring to **Creating a Mirror Filter**.

6. Click **Next**.

   The **Associate Mirror Sources** page is displayed.

7. In the mirror source list, select mirror sources.

   When associating mirror sources with a mirror session, learn about the constraints on mirror sources in **Table 13-5**.

8. Click **Next**.

   The **Associate Mirror Target** page is displayed.

9. In the mirror target list, select a mirror target.
   - A mirror target is a network interface of an ECS or a load balancer, which is used to receive mirrored traffic.
   - Each mirror session can be associated with only one mirror target.

     For details about the constraints on mirror targets, see **Table 13-6**.

10. Click **Next**.

    The **Confirm** page is displayed.

11. After confirming that the configuration is correct, click **Create Now**.

    You can view the created mirror session in the mirror session list.

13.3.2 Enabling or Disabling a Mirror Session

Scenarios

You can enable or disable a mirror session.

- If the mirror session is disabled, traffic of mirror sources cannot be monitored.
- If the mirror session is enabled, traffic of mirror sources can be monitored.

Procedure

1. Go to the **mirror session list page**.
2. In the mirror session list, locate the row that contains the mirror session and click **Enable** or **Disable** in the **Operation** column.

   A confirmation dialog box is displayed.

3. Click **OK**.

13.3.3 Associating Mirror Sources with a Mirror Session

Scenarios

You can associate mirror sources with a mirror session.

Constraints

When associating mirror sources with a mirror session, learn about the constraints on mirror sources in **Table 13-5**.

Procedure

1. Go to the **mirror session list page**.
2. In the mirror session list, locate the row that contains the mirror session and click its name with a hyperlink.

   The **Basic Information** page is displayed.
3. Click the **Mirror Sources** tab. In the upper left corner of the mirror source list, click **Associate**.

   The **Associate Mirror Sources** dialog box is displayed.
4. In the mirror source list, select mirror sources and click **OK**.

   In the mirror source list, you can view the associated mirror sources.

13.3.4 Disassociating Mirror Sources from a Mirror Session

Scenarios

You can disassociate mirror sources from a mirror session.

Procedure

1. Go to the **mirror session list page**.
2. In the mirror session list, locate the row that contains the mirror session and click its name with a hyperlink.

   The **Basic Information** page is displayed.
3. Click the **Mirror Sources** tab. In the mirror source list, locate the row that contains the mirror source and click **Disassociate** in the **Operation** column.

   A confirmation dialog box is displayed.
4. Click **OK**.

   After the disassociation is successful, the mirror source list is displayed.

13.3.5 Changing the Mirror Filter for a Mirror Session

Scenarios

A mirror session can be associated with only one mirror filter. If the current mirror filter cannot meet your requirements, you can change one.

Procedure

1. Go to the **mirror session list page**.
2. In the mirror session list, locate the row that contains the mirror session and choose **More** > **Change Mirror Filter** in the **Operation** column.

   The **Change Mirror Filter** dialog box is displayed.

3. In the mirror filter list, select a mirror filter and click **OK**.

   After the change, you can see that the new mirror filter in the **Mirror Filter** column of the mirror session list.

   If there is no mirror filter you want, create one by referring to **Creating a Mirror Filter**.

13.3.6 Changing the Mirror Target of a Mirror Session

Scenarios

A mirror session can be associated with only one mirror target. You can change the mirror target of a mirror session.

Constraints

- A mirror target is a network interface of an ECS or a load balancer, which is used to receive mirrored traffic.
- Each mirror session can be associated with only one mirror target.

  For details about the constraints on mirror targets, see **Table 13-6**.

Procedure

1. Go to the **mirror session list page**.
2. In the mirror session list, locate the row that contains the mirror session and choose **More** > **Change Mirror Target** in the **Operation** column.

   The **Change Mirror Target** dialog box is displayed.
3. In the mirror target list, select a mirror target, and click **OK**.

   After the change, you can see that the new mirror target in the **Mirror Target** column of the mirror session list.

13.3.7 Modifying the Basic Information About a Mirror Session

You can modify the basic information about a mirror session, including its name and description.

Procedure

1. Go to the **mirror session list page**.
2. In the mirror session list, locate the row that contains the mirror session and click **Modify** in the **Operation** column.

   The **Modify Mirror Session** dialog box is displayed.
3. Modify the parameters as prompted.

Table 13-21 Parameters that can be modified

Parameter	Description	Example Value
Name	Mandatory Enter a different name for the mirror session. The name: • Can contain 1 to 64 characters. • Can contain letters, digits, underscores (_), hyphens (-), and periods (.).	mirror-session-01
Description	Optional Enter a description for the mirror session in the text box as required.	-

4. Click **OK** to save the modification.

13.3.8 Viewing the Details About a Mirror Session

Scenarios

You can view the following information about a mirror session:

- Basic information, such as name, priority, and description
- Mirror filter
- Mirror sources, such as the private IP addresses, attached instances, and security groups of elastic network interfaces
- Mirror target, such as an ECS's network interface or a load balancer

Procedure

1. Go to the **mirror session list page**.
2. In the mirror session list, you can view the mirror session name, mirror filter, and mirror target.
3. In the mirror session list, locate the row that contains the mirror session and click its name with a hyperlink.

   The **Basic Information** page is displayed.
4. View information about the mirror session on different tab pages.
   - On the **Basic Information** tab page, view mirror session information, such as name, priority, and description.
   - On the **Mirror Sources** tab page, view the private IP addresses, attached instances, and security groups of elastic network interfaces

13.3.9 Deleting a Mirror Session

Scenarios

If a mirror session is no longer required, you can delete it.

Procedure

1. Go to the **mirror session list page**.
2. In the mirror session list, locate the row that contains the mirror session and choose **More** > **Delete** in the **Operation** column.

   A confirmation dialog box is displayed.
3. Confirm the information and click **OK**.

   A deleted mirror session cannot be recovered.

13.4 Traffic Mirroring Example Scenarios

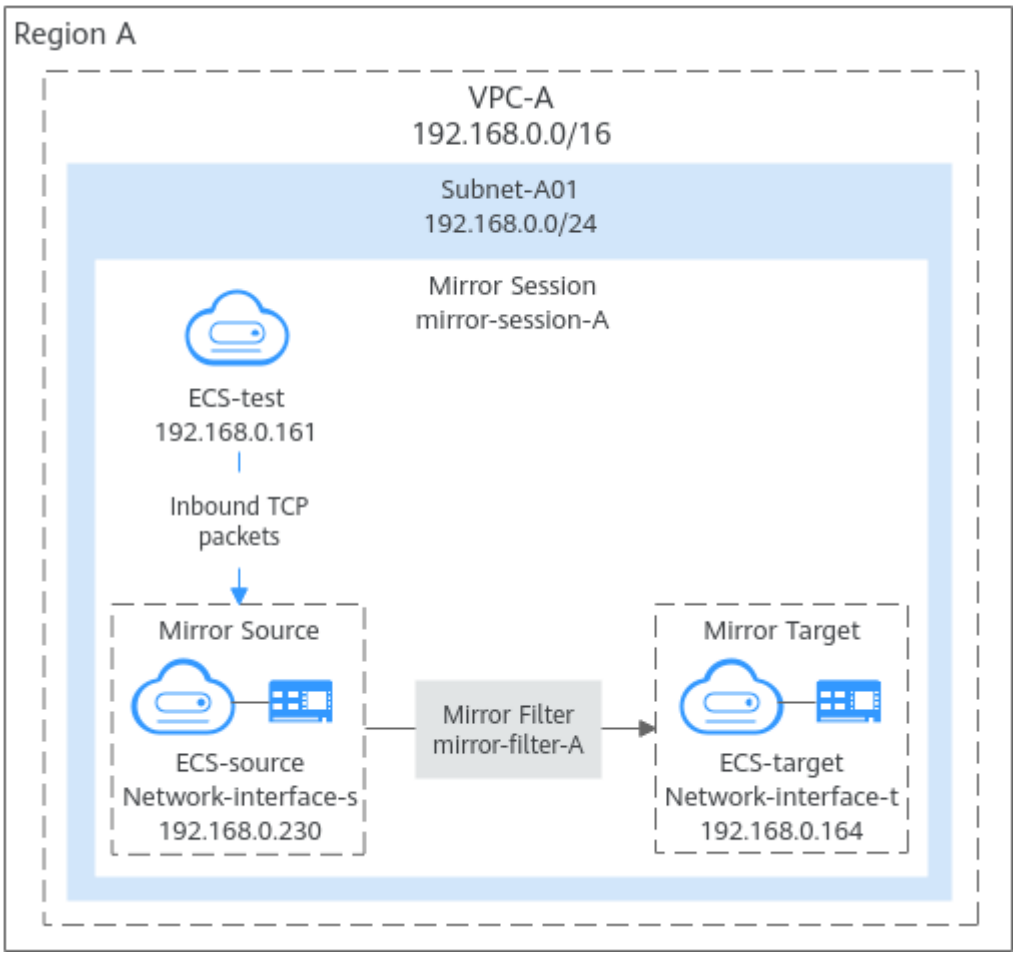
13.4.1 Mirroring Inbound TCP Traffic to a Single Network Interface

Solution Architecture

To mirror inbound TCP traffic from a mirror source (network interface) to a single mirror target (network interface), you can refer to the configurations in this section. In **Figure 13-6**, when **ECS-test** accesses **ECS-source**, you can create a mirror session to mirror inbound TCP traffic on **ECS-source** to **ECS-target**.

- Set the mirror source to **Network-interface-s** on **ECS-source**. The inbound TCP traffic on this network interface needs to be mirrored.
- Set the mirror target to **Network-interface-t** on **ECS-target**. The inbound TCP traffic on **network-interface-s** is mirrored to **network-interface-t**.
- Associate the mirror filter that has a rule for accepting inbound TCP traffic with the mirror session.

Figure 13-6 Mirroring inbound TCP traffic



Notes and Constraints

See **Notes and Constraints**.

Resource Planning

In this example, the VPC, subnet, EIP, and ECSs must be in the same region but can be in different AZs.

NOTE

The following resource details are only for your reference. You can modify them if needed.

Table 13-22 Resource details for mirroring inbound TCP traffic

Resource	Quantity	Description
VPC and subnet	VPC: 1 Subnet: 1	• **Name**: Set it as needed. In this example, **VPC-A** is used. • **VPC IPv4 CIDR Block**: Set it as needed. In this example, **192.168.0.0/16** is used. • **Subnet Name**: Set it as needed. In this example, **Subnet-A01** is used. • **Subnet IPv4 CIDR Block**: Set it as needed. In this example, **192.168.0.0/24** is used.

Resource	Quantity	Description
ECS	3	Configure the ECSs as follows: • **ECS Name**: Set it as needed. In this example, the ECSs are named **ECS-source**, **ECS-target**, and **ECS-test**. • **ECS Type**: In this example, the type of **ECS-source** is **General computing-plus c7t**. Currently, only network interfaces of ECSs of certain types can be used as mirror sources. For details, see **Notes and Constraints**. There are no constraints on the type of **ECS-target** and **ECS-test**. • **Image**: Set it as needed. In this example, public image **Huawei Cloud EulerOS 2.0 Standard 64 bit** is used. • **System Disk**: In this example, a general-purpose SSD disk of 40 GiB is used. • **Data Disk**: Set it as needed. In this example, no data disk is used. • **Network** – **VPC**: Select a VPC. In this example, **VPC-A** is used. – **Subnet**: Select a subnet. In this example, **Subnet-A01** is used. • **Security Group**: In this example, the three ECSs are associated with the same security group (**Sg-X**). Ensure that all rules in **Table 13-23** are added. If the ECSs are associated with different security groups, you also need to add additional rules. – If **ECS-test** is associated with **Sg-X** and **ECS-source** is associated with **Sg-A**, add the rule in **Table 13-24** to **Sg-A** to allow traffic from **ECS-test**. – If **ECS-source** is associated with **Sg-A** but **ECS-target** is associated with **Sg-B**, add the rule in **Table 13-25** to **Sg-B** to allow UDP packets encapsulated by the mirror source to access the mirror target over port 4789. • **EIP**: Select **Not required**. • Private IP address: In this example, use **192.168.0.230** for **ECS-source**, **192.168.0.164** for **ECS-target**, and **192.168.0.161** for **ECS-test**.
EIP	1	• **Billing Mode**: Set it as needed. In this example, **Pay-per-use** is used. • **EIP Name**: Set it as needed. In this example, **EIP-A** is used. • EIP: The EIP is randomly assigned. In this example, **124.X.X.187** is used.

Resource	Quantity	Description
Mirror filter	1	• **Name**: Set it as needed. In this example, **mirror-filter-A** is used. • Inbound rule: Add the inbound rule in **Table 13-26**. This rule allows TCP packets from **ECS-test** to **ECS-source** over port 1234 to be mirrored.
Mirror session	1	• **Basic Information** – **Name**: Set it as needed. In this example, **mirror-session-A** is used. – **Priority**: Set it as needed. In this example, **1** is used. – **VNI**: Set it as needed. In this example, **1** is used. – **Packet Length**: Set it as needed. In this example, **96** is used. – **Mirror Session**: Enable it to mirror the traffic from the mirror source. • **Associate Mirror Filter**: Set it as needed. In this example, **mirror-filter-A** is used. • **Associate Mirror Sources**: Set it as needed. In this example, the private IP address (192.168.0.230) of the network interface of **ECS-source** is used. • **Associate Mirror Target** – **Type**: **Network interface** – Network interface: Set it as needed. In this example, the private IP address (192.168.0.164) of the network interface of **ECS-target** is used.

Table 13-23 Security group **Sg-X** rules

Direction	Action	Type	Protocol & Port	Source/ Destination	Description
Inbound	Allow	IPv4	TCP: 22	Source: 0.0.0.0/0	Allows remote logins to Linux ECSs over SSH port 22.
Inbound	Allow	IPv4	TCP: 3389	Source: 0.0.0.0/0	Allows remote logins to Windows ECSs over RDP port 3389.
Inbound	Allow	IPv4	All	Source: current security group (**Sg-X**)	Allows the ECSs in this security group to communicate with each other using IPv4 addresses.

Direction	Action	Type	Protocol & Port	Source/ Destination	Description
Inbound	Allow	IPv6	All	Source: current security group (**Sg-X**)	Allows the ECSs in this security group to communicate with each other using IPv6 addresses.
Outbound	Allow	IPv4	All	Destination: 0.0.0.0/0	Allows ECSs in this security group to access the Internet using IPv4 addresses.
Outbound	Allow	IPv6	All	Destination: ::/0	Allows ECSs in this security group to access the Internet using IPv6 addresses.

NOTICE

If the source of an inbound rule is set to 0.0.0.0/0, all external IP addresses are allowed to remotely log in to your cloud server. Exposing port 22 or 3389 to the public network will leave your instances vulnerable to network risks. To address this issue, set the source to a trusted IP address, for example, the IP address of your local PC.

Table 13-24 Security group **Sg-A** rules

Direction	Action	Type	Protocol & Port	Source	Description
Inbound	Allow	IPv4	TCP: 1234	Private IP address of the ECS that accesses the mirror source. In this example, the private IP address of **ECS-test** is used: 192.168.0.161/32	Allows TCP packets from **ECS-test** to **ECS-source** over port 1234.

Table 13-25 Security group **Sg-B** rules

Direction	Action	Type	Protocol & Port	Source	Description
Inbound	Allow	IPv4	UDP: 4789	The private IP address of mirror source **ECS-source**: 192.168.0.230/32	Allows UDP packets encapsulated by **ECS-source** to access **ECS-target** over port 4789.

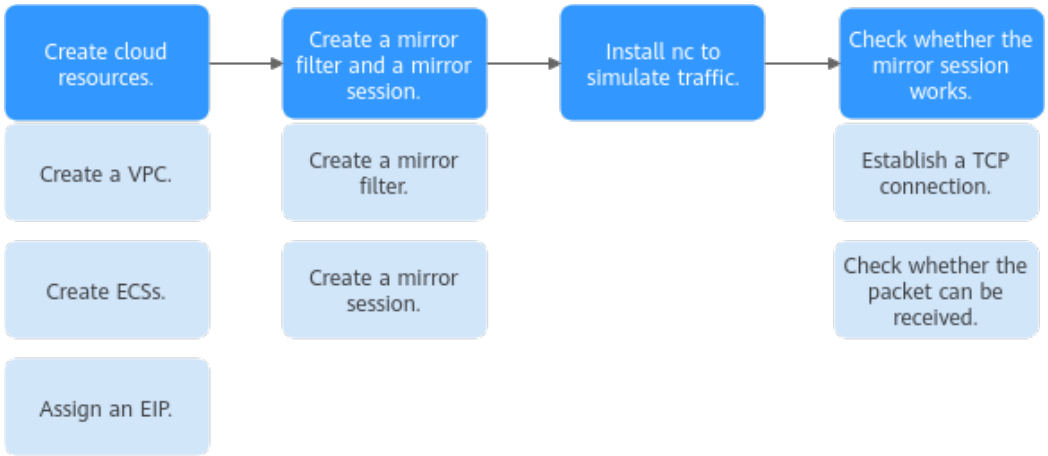
Table 13-26 Inbound rules of the mirror filter

Direction	Priority	Protocol	Action	Type	Source	Source Port Range	Destination	Destination Port Range
Inbound	1	TCP	Accept	IPv4	The private IP address of **ECS-test**: 192.168.0.161/32	All	The private IP address of **ECS-source**: 192.168.0.230/32	Port of **ECS-source**: 1234-1234

Procedure

Figure 13-7 shows the procedure required to mirror inbound TCP traffic to a single network interface.

Figure 13-7 Mirroring inbound TCP traffic



Step 1: Create Cloud Resources

1. Create a VPC and subnet.

For details, see **Creating a VPC and Subnet**.

2. Create three ECSs.

   For details, see **Purchasing a Custom ECS**.

3. Assign an EIP.

   For details, see **Assigning an EIP**.

Step 2: Create a Mirror Filter and a Mirror Session

1. Create a mirror filter.

   For details, see **Creating a Mirror Filter**.

2. Create a mirror session, and associate the mirror filter, mirror source, and mirror target with this mirror session.

   For details, see **Creating a Mirror Session**.

Step 3: Install Netcat (nc) to Simulate Traffic

The nc utility reads and writes data across network connections using TCP or UDP. It is usually used to test ports for accessibility. You need to install nc on both **ECS-source** and **ECS-test**.

1. Install nc on **ECS-source**.

   a. Bind the EIP to **ECS-source** to connect to the Internet for downloading the nc utility.

      For details, see **Binding an EIP to an ECS**.

   b. Remotely log in to **ECS-source**.

      For details, see **How Do I Log In to My ECS?**

   c. Run the following commands in sequence to install nc:

      sudo yum update

      Information similar to the following is displayed:

```
[root@ecs-source ~]# sudo yum update
HCE 2.0
base
         55 MB/s | 6.1 MB     00:00
HCE 2.0
updates
           98 MB/s |  14 MB     00:00
Last metadata expiration check: 0:00:01 ago on Tue 10 Sep 2024 05:54:28 PM CST.
Dependencies resolved.
Nothing to do.
Complete!
```

      sudo yum install nc

      If information similar to the following is displayed, enter **y** as prompted and press **Enter**:

```
[root@ecs-source ~]# sudo yum install nc
Last metadata expiration check: 0:00:12 ago on Tue 10 Sep 2024 05:54:28 PM CST.
Dependencies resolved.
...
Install  2 Packages

Total download size: 6.1 M
Installed size: 25 M
Is this ok [y/N]: y
Downloading Packages:
...
```



```
Importing GPG key 0xA8DEF926:
 Userid     : "HCE <support@huaweicloud.com>"
 Fingerprint: C1BA 9CD4 9D03 A206 E241 F176 28DA 5B77 A8DE F926
 From       : http://repo.huaweicloud.com/hce/2.0/updates/RPM-GPG-KEY-HCE-2
Is this ok [y/N]: y
...
Installed:
  libssh2-1.10.0-2.r10.hce2.x86_64
nmap-2:7.92-2.r4.hce2.x86_64

Complete!
```

d. Unbind the EIP from **ECS-source** after nc is installed.

For details, see **Unbinding an EIP**.

2. Repeat **1.a** to **1.d** on **ECS-test**.
3. Release the EIP.

For details, see **Unbinding an EIP**. If you do not release the EIP, the EIP will continue to be billed.

Step 3: Check Whether the Mirror Session Works

1. Establish a TCP connection between **ECS-source** and **ECS-test**.

Send TCP packets from **ECS-test** to **ECS-source** and check whether **ECS-source** can receive the packets.

a. Run the following command on **ECS-source** to listen to port 1234:

nc -l *<listening-port-of-mirror-source-ECS-source>*

Example command:

nc -l 1234

If the command output is empty, the port is opened for listening.

b. Run the following command on **ECS-test** to establish a TCP connection between **ECS-source** and **ECS-test**:

nc *<private-IP-address-of-mirror-source-ECS-source> <listening-port-of-mirror-source-ECS-source>*

Example command:

nc 192.168.0.230 1234

The command output is empty. Enter any information (for example, **hello**) on **ECS-test** and press **Enter** to check whether the TCP connection is successfully established.

```
[root@ecs-test ~]# nc 192.168.0.230 1234
hello
```

c. Check whether **ECS-source** can receive information from **ECS-test**.

If information similar to the following is displayed, the TCP connection is successfully established.

```
[root@ecs-source ~]# nc -l 1234
hello
```

2. Check whether the inbound packet on **ECS-source** can be mirrored to **ECS-target**.

When **ECS-test** sends a TCP packet to **ECS-source**, run **tcpdump** to check whether **ECS-target** can receive the packet. If **ECS-target** receives the packet, the mirror session works.

a. Remotely log in to **ECS-target**.

For details, see **How Do I Log In to My ECS?**

b. Run the following command on **ECS-target** to view its network interface name:

ifconfig

Information similar to the following is displayed. In this example, the network interface of the mirror target is **eth0**.

```
[root@ecs-target ~]# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.0.164  netmask 255.255.255.0  broadcast 192.168.0.255
        inet6 fe80::f816:3eff:fe7e:d67a  prefixlen 64  scopeid 0x20<link>
        ether fa:16:3e:7e:d6:7a  txqueuelen 1000  (Ethernet)
        RX packets 29043  bytes 32268398 (30.7 MiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 13811  bytes 3961116 (3.7 MiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
...
```

c. Run the following command on **ECS-target** to check whether it can receive packets:

tcpdump -i *<network-interface-name-of-the-mirror-target>* **udp port 4789 -nne**

Example command:

tcpdump -i eth0 udp port 4789 -nne

Information similar to the following is displayed:

```
[root@ecs-target ~]# tcpdump -i eth0 udp port 4789 -nne
dropped privs to tcpdump
tcpdump: verbose output suppressed, use -v[v]... for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), snapshot length 262144 bytes
```

d. Enter any information (for example, **12345**) on **ECS-test** and press **Enter** to send a TCP packet to **ECS-source**.

Information similar to the following is displayed:

```
[root@ecs-test ~]# nc 192.168.0.230 1234
hello
12345
```

e. Check whether **ECS-source** can receive "12345" from **ECS-test**.

If information similar to the following is displayed, **ECS-source** can receive "12345" from **ECS-test**:

```
[root@ecs-source ~]# nc -l 1234
hello
12345
```

f. Check whether **ECS-target** can receive the packet.

Information similar to the following is displayed. You can view the packet containment **12345** sent by **ECS-test** after running **tcpdump**. **vni 1** is the identifier of the mirror session, indicating that **ECS-target** can receive the packet through the mirror session. The packet content has two parts. For details, see **Table 13-27**.

```
[root@ecs-target ~]# tcpdump -i eth0 udp port 4789 -nne
dropped privs to tcpdump
tcpdump: verbose output suppressed, use -v[v]... for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), snapshot length 262144 bytes
19:12:25.839624 fa:16:3e:d1:6b:5d > fa:16:3e:7e:d6:7a, ethertype IPv4 (0x0800), length 122:
192.168.0.230.32838 > 192.168.0.164.4789: VXLAN, flags [I] (0x08), vni 1
fa:16:3e:7e:d6:77 > fa:16:3e:7e:d6:bc, ethertype IPv4 (0x0800), length 72: 192.168.0.161.38944 >
192.168.0.230.1234: Flags [P.], seq 2063075043:2063075049, ack 1116663338, win 502, options
[nop,nop,TS val 969673134 ecr 605179348], length 6
```


Table 13-27 Packet description

Packet Example	Packet Description
19:12:25.839624 fa:16:3e:d1:6b:5d > fa:16:3e:7e:d6:7a, ethertype IPv4 (0x0800), length 122: 192.168.0.230.32838 > 192.168.0.164.4789: VXLAN, flags [I] (0x08), vni 1	VXLAN packet encapsulated by Traffic Mirroring. Packet format: <Timestamp><SMacAddr><DMacAddr><EthernetType><Length><Sip><Sport><Dip><Dport><VXLAN Flags><VNI> Fields in the encapsulated packet: • Timestamp: Time when a packet is obtained. It is generated by tcpdump. • SMacAddr: MAC address of the source instance of VXLAN packets. In this example, it is the MAC address of the gateway instance. • DMacAddr: MAC address of the target instance of VXLAN packets. In this example, it is the MAC address of the mirror target instance. • EthernetType: indicates the Ethernet type of a packet. 0x0800 indicates that the protocol is IPv4. • Length: packet length • Sip: Mirror source address • Sport: Mirror source port • Dip: Mirror target address • Dport: Mirror target port, which is usually port 4789 that receives VXLAN packets • VXLAN Flags: The value is usually 0x08, indicating a VXLAN packet. • VNI: VXLAN network identifier of a mirror session
fa:16:3e:7e:d6:77 > fa:16:3e:7e:d6:bc, ethertype IPv4 (0x0800), length 72: 192.168.0.161.38944 > 192.168.0.230.1234: Flags [P.], seq 2063075043:2063075049, ack 1116663338, win 502, options [nop,nop,TS val 969673134 ecr 605179348], length 6	Original packet The original packet field is general network knowledge and is not described in detail herein.

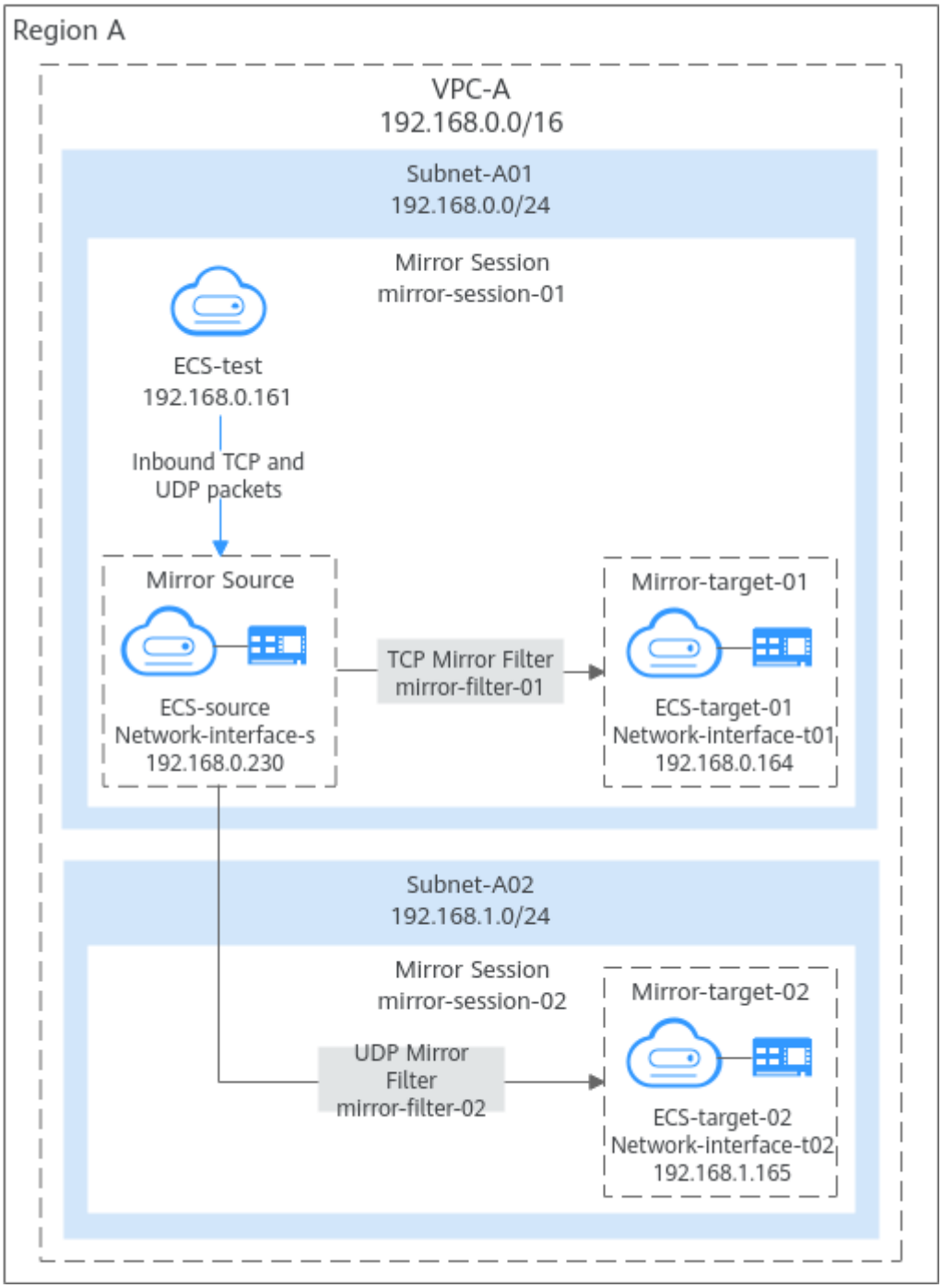
13.4.2 Mirroring Inbound TCP and UDP Traffic to Multiple Network Interfaces

Solution Architecture

To mirror inbound TCP and UDP traffic from a mirror source (network interface) to different mirror targets (network interfaces), you can refer to the configurations in this section. In **Figure 13-8**, when **ECS-test** accesses **ECS-source**, the inbound TCP traffic on **ECS-source** needs to be mirrored to **ECS-target-01** and the inbound UDP traffic on **ECS-source** needs to be mirrored to **ECS-target-02**. Each mirror session can only be associated with one mirror target, so you need to create two mirror sessions.

- **mirror-session-01**:
  - Set the mirror source to **Network-interface-s** on **ECS-source**. The inbound TCP traffic on this network interface needs to be mirrored.
  - Set the mirror target to **Network-interface-t01** on **ECS-target-01**. The inbound TCP traffic on **Network-interface-s** is mirrored to **Network-interface-t01**.
  - Associate **mirror-filter-01** that has a rule for accepting inbound TCP traffic with **mirror-session-01**.
- **mirror-session-02**:
  - Set the mirror source to **Network-interface-s** of **ECS-source**, indicating that the inbound UDP traffic on this network interface needs to be mirrored.
  - Set the mirror target to **Network-interface-t02** of **ECS-target-02**, indicating that the inbound UDP traffic on network-interface-s is mirrored to **Network-interface-t02**.
  - Associate **mirror-filter-02** that has a rule for accepting inbound UDP traffic with **mirror-session-02**.

Figure 13-8 Mirroring inbound TCP and UDP traffic



Notes and Constraints

See **Notes and Constraints**.

Resource Planning

In this example, the VPCs, subnets, EIP, and ECSs must be in the same region but can be in different AZs.

NOTE

The following resource details are only for your reference. You can modify them if needed.

Table 13-28 Resource details for mirroring inbound TCP and UDP traffic

Resource	Quantity	Description
VPC and subnet	VPC: 1 Subnet: 2	• **Name**: Set it as needed. In this example, **VPC-A** is used. • **VPC IPv4 CIDR Block**: Set it as needed. In this example, **192.168.0.0/16** is used. • **Subnet Name**: Set it as needed. In this example, **Subnet-A01** and **Subnet-A02** are used. • **Subnet IPv4 CIDR Block**: Set it as needed. In this example, the CIDR block of **Subnet-A01** is **192.168.0.0/24** and that of **Subnet-A02** is **192.168.1.0/24**.

Resource	Quantity	Description
ECS	4	Configure the ECSs as follows: • **ECS Name**: Set it as needed. In this example, the ECSs are named **ECS-source**, **ECS-target-01**, **ECS-target-02**, and **ECS-test**. • **ECS Type**: In this example, the type of **ECS-source** is **General computing-plus c7t**. Currently, only network interfaces of ECSs of certain types can be used as mirror sources. For details, see **Notes and Constraints**. There are no constraints on the type of other ECSs. • **Image**: Set it as needed. In this example, public image **Huawei Cloud EulerOS 2.0 Standard 64 bit** is used. • **System Disk**: In this example, a general-purpose SSD disk of 40 GiB is used. • **Data Disk**: Set it as needed. In this example, no data disk is used. • **Network** – **VPC**: Select a VPC. In this example, **VPC-A** is used. – **Subnet**: Select a subnet. In this example, the subnet of **ECS-source**, **ECS-target-01** and **ECS-test** is **Subnet-A01**, and that of **ECS-target-02** is **Subnet-A02**. • **Security Group**: In this example, the four ECSs are associated with the same security group (**Sg-X**). Ensure that all rules in **Table 13-29** are added. If the ECSs are associated with different security groups, you also need to add additional rules. – If **ECS-test** is associated with **Sg-X** and **ECS-source** is associated with **Sg-A**, add the rules in **Table 13-30** to **Sg-A** to allow traffic from **ECS-test**. – If **ECS-source** is associated with **Sg-A** and **ECS-target-01** is associated with **Sg-B**, add the rule in **Table 13-31** to **Sg-B** to allow UDP packets encapsulated by the mirror source to access the mirror target over port 4789. The same applies to **ECS-target-02**. • **EIP**: Select **Not required**. • Private IP address: In this example, use **192.168.0.230** for **ECS-source**, **192.168.0.164** for **ECS-target-01**, **192.168.1.165** for **ECS-target-02**, and **192.168.0.161** for **ECS-test**.

Resource	Quantity	Description
EIP	1	• **Billing Mode**: Set it as needed. In this example, **Pay-per-use** is used. • **EIP Name**: Set it as needed. In this example, **EIP-A** is used. • EIP: The EIP is randomly assigned. In this example, **124.X.X.187** is used.
Mirror filter	2	• One mirror filter for accepting TCP traffic: – **Name**: Set it as needed. In this example, **mirror-filter-01** is used. – Inbound rule: Add the inbound rule in **Table 13-32**. This rule allows TCP packets from **ECS-test** to **ECS-source** over port 1234 to be mirrored. • One mirror filter for accepting UDP traffic: – **Name**: Set it as needed. In this example, **mirror-filter-02** is used. – Inbound rule: Add the inbound rule in **Table 13-32**. This rule allows UDP packets from **ECS-test** to **ECS-source** over port 1235 to be mirrored.

Resource	Quantity	Description
Mirror session	2	One mirror session for accepting TCP traffic: • **Basic Information**: – **Name**: Set it as needed. In this example, **mirror-session-01** is used. – **Priority**: Set it as needed. In this example, **1** is used. – **VNI**: Set it as needed. In this example, **1** is used. – **Packet Length**: Set it as needed. In this example, **96** is used. – **Mirror Session**: Enable it to mirror the traffic from the mirror source. • **Associate Mirror Filter**: Set it as needed. In this example, **mirror-filter-01** is used. • **Associate Mirror Sources**: Set it as needed. In this example, the private IP address (192.168.0.230) of the network interface of **ECS-source** is used. • **Associate Mirror Target** – **Type**: **Network interface** – Network interface: Set it as needed. In this example, the private IP address (192.168.0.164) of the network interface of **ECS-target-01** is used. One mirror session for accepting UDP traffic: • **Basic Information** – **Name**: Set it as needed. In this example, **mirror-session-02** is used. – **Priority**: Set it as needed. In this example, **2** is used. – **VNI**: Set it as needed. In this example, **2** is used. – **Packet Length**: Set it as needed. In this example, **96** is used. – **Mirror Session**: Enable it to mirror the traffic from the mirror source. • **Associate Mirror Filter**: Set it as needed. In this example, **mirror-filter-02** is used. • **Associate Mirror Sources**: Set it as needed. In this example, the private IP address (192.168.0.230) of the network interface of **ECS-source** is used. • **Associate Mirror Target** – **Type**: **Network interface** – Network interface: Set it as needed. In this example, the private IP address (192.168.1.165) of the network interface of **ECS-target-02** is used.

Table 13-29 Security group **Sg-X** rules

Direction	Action	Type	Protocol & Port	Source/ Destination	Description
Inbound	Allow	IPv4	TCP: 22	Source: 0.0.0.0/0	Allows remote logins to Linux ECSs over SSH port 22.
Inbound	Allow	IPv4	TCP: 3389	Source: 0.0.0.0/0	Allows remote logins to Windows ECSs over RDP port 3389.
Inbound	Allow	IPv4	All	Source: current security group (**Sg-X**)	Allows the ECSs in this security group to communicate with each other using IPv4 addresses.
Inbound	Allow	IPv6	All	Source: current security group (**Sg-X**)	Allows the ECSs in this security group to communicate with each other using IPv6 addresses.
Outbound	Allow	IPv4	All	Destination: 0.0.0.0/0	Allows ECSs in this security group to access the Internet using IPv4 addresses.
Outbound	Allow	IPv6	All	Destination: ::/0	Allows ECSs in this security group to access the Internet using IPv6 addresses.

NOTICE

If the source of an inbound rule is set to 0.0.0.0/0, all external IP addresses are allowed to remotely log in to your cloud server. Exposing port 22 or 3389 to the public network will leave your instances vulnerable to network risks. To address this issue, set the source to a trusted IP address, for example, the IP address of your local PC.

Table 13-30 Security group **Sg-A** rules

Direction	Action	Type	Protocol & Port	Source	Description
Inbound	Allow	IPv4	TCP: 1234	Private IP address of the ECS that accesses the mirror source. In this example, the private IP address of **ECS-test** is used: 192.168.0.161/32	Allows TCP packets from **ECS-test** to **ECS-source** over port 1234.
Inbound	Allow	IPv4	UDP: 1235	Private IP address of the ECS that accesses the mirror source. In this example, the private IP address of **ECS-test** is used: 192.168.0.161/32	Allows UDP packets from **ECS-test** to **ECS-source** over port 1235.

Table 13-31 Security group **Sg-B** rule

Direction	Action	Type	Protocol & Port	Source	Description
Inbound	Allow	IPv4	UDP: 4789	The private IP address of mirror source **ECS-source**: 192.168.0.230/32	Allows UDP packets encapsulated by **ECS-source** to access **ECS-target-01** over port 4789.

Table 13-32 Inbound rules of the mirror filter

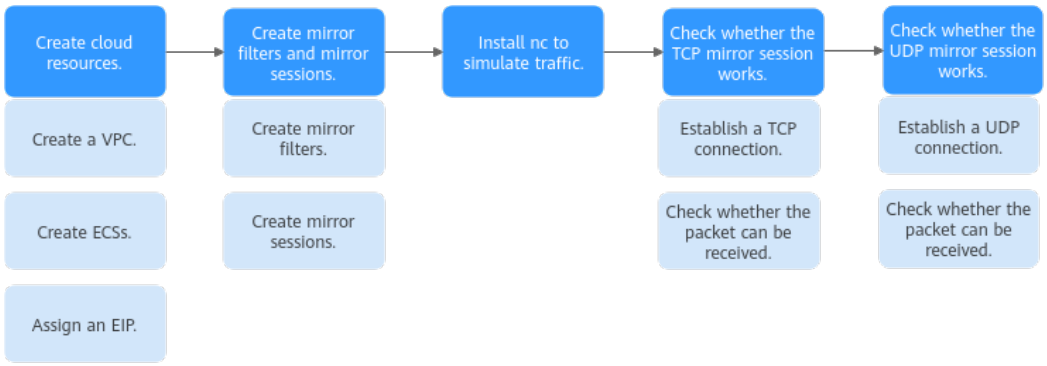
Name	Direction	Priority	Protocol	Action	Type	Source	Source Port Range	Destination	Destination Port Range
mirror-filter-01	Inbound	1	TCP	Accept	IPv4	The private IP address of **ECS-test**: 192.168.0.161/32	All	The private IP address of **ECS-source**: 192.168.0.230/32	Port of **ECS-source**: 1234-1234

Name	Direction	Priority	Protocol	Action	Type	Source	Source Port Range	Destination	Destination Port Range
mirror-filter-02	Inbound	1	UDP	Accept	IPv4	The private IP address of **ECS-test**: 192.168.0.161/32	All	The private IP address of **ECS-source**: 192.168.0.230/32	In this example, port 1235 of **ECS-source** is used. 1235-1235

Procedure

Figure 13-9 shows the procedure required to mirror inbound TCP and UDP traffic to multiple network interfaces.

Figure 13-9 Mirroring inbound TCP and UDP traffic



Step 1: Create Cloud Resources

1. Create a VPC with two subnets.
   For details, see **Creating a VPC and Subnet**.
2. Create four ECSs.
   For details, see **Purchasing a Custom ECS**.
3. Assign an EIP.
   For details, see **Assigning an EIP**.

Step 2: Create Mirror Filters and Mirror Sessions

1. Create two mirror filters.
   For details, see **Creating a Mirror Filter**.
2. Create two mirror sessions, and associate the mirror filters, mirror sources, and mirror targets with the mirror sessions.

For details, see **Creating a Mirror Session**.

Step 3: Install Netcat (nc) to Simulate Traffic

The nc utility reads and writes data across network connections using TCP or UDP. It is usually used to test ports for accessibility. You need to install nc on both **ECS-source** and **ECS-test**.

1. Install nc on **ECS-source**.
   a. Bind the EIP to **ECS-source** to connect to the Internet for downloading the nc utility.

      For details, see **Binding an EIP to an ECS**.
   b. Remotely log in to **ECS-source**.

      For details, see **How Do I Log In to My ECS?**
   c. Run the following commands in sequence to install nc:

      sudo yum update

      Information similar to the following is displayed:

```
[root@ecs-source ~]# sudo yum update
HCE 2.0
base
         55 MB/s | 6.1 MB     00:00
HCE 2.0
updates
           98 MB/s |  14 MB     00:00
Last metadata expiration check: 0:00:01 ago on Tue 10 Sep 2024 05:54:28 PM CST.
Dependencies resolved.
Nothing to do.
Complete!
```

      sudo yum install nc

      If information similar to the following is displayed, enter **y** as prompted and press **Enter**:

```
[root@ecs-source ~]# sudo yum install nc
Last metadata expiration check: 0:00:12 ago on Tue 10 Sep 2024 05:54:28 PM CST.
Dependencies resolved.
...
Install  2 Packages

Total download size: 6.1 M
Installed size: 25 M
Is this ok [y/N]: y
Downloading Packages:
...
Importing GPG key 0xA8DEF926:
 Userid     : "HCE <support@huaweicloud.com>"
 Fingerprint: C1BA 9CD4 9D03 A206 E241 F176 28DA 5B77 A8DE F926
 From       : http://repo.huaweicloud.com/hce/2.0/updates/RPM-GPG-KEY-HCE-2
Is this ok [y/N]: y
...
Installed:
  libssh2-1.10.0-2.r10.hce2.x86_64
nmap-2:7.92-2.r4.hce2.x86_64

Complete!
```

   d. Unbind the EIP from **ECS-source** after nc is installed.

      For details, see **Unbinding an EIP**.
2. Repeat **1.a** to **1.d** on **ECS-test**.
3. Release the EIP.

For details, see **Unbinding an EIP**. If you do not release the EIP, the EIP will continue to be billed.

Step 4: Check Whether the TCP Mirror Session Works

1. Establish a TCP connection between **ECS-source** and **ECS-test**.

   Send TCP packets from **ECS-test** to **ECS-source** and check whether **ECS-source** can receive the packets.

   a. Run the following command on **ECS-source** to listen to port 1234:

      nc -l *<listening-port-of-mirror-source-ECS-source>*

      Example command:

      nc -l 1234

      If the command output is empty, the port is opened for listening.

   b. Run the following command on **ECS-test** to establish a TCP connection between **ECS-source** and **ECS-test**:

      nc *<private-IP-address-of-mirror-source-ECS-source> <listening-port-of-mirror-source-ECS-source>*

      Example command:

      nc 192.168.0.230 1234

      The command output is empty. Enter any information (for example, **hello**) on **ECS-test** and press **Enter** to check whether the TCP connection is successfully established.

      ```
      [root@ecs-test ~]# nc 192.168.0.230 1234
      hello
      ```

   c. Check whether **ECS-source** can receive information from **ECS-test**.

      If information similar to the following is displayed, the TCP connection is successfully established.

      ```
      [root@ecs-source ~]# nc -l 1234
      hello
      ```

2. Check whether the inbound TCP packets on **ECS-source** can be mirrored to **ECS-target-01**.

   When **ECS-test** sends a TCP packet to **ECS-source**, run **tcpdump** to check whether **ECS-target-01** can receive the packet. If **ECS-target-01** receives the packet, the mirror session works.

   a. Remotely log in to **ECS-target-01**.

      For details, see **How Do I Log In to My ECS?**

   b. Run the following command on **ECS-target-01** to view its network interface name:

      ifconfig

      Information similar to the following is displayed. In this example, the network interface of the mirror target is **eth0**.

      ```
      [root@ecs-target-01 ~]# ifconfig
      eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
            inet 192.168.0.164  netmask 255.255.255.0  broadcast 192.168.0.255
            inet6 fe80::f816:3eff:fe7e:d67a  prefixlen 64  scopeid 0x20<link>
            ether fa:16:3e:7e:d6:7a  txqueuelen 1000  (Ethernet)
            RX packets 283560  bytes 116380316 (110.9 MiB)
            RX errors 0  dropped 0  overruns 0  frame 0
            TX packets 276486  bytes 104575280 (99.7 MiB)
      ```



```
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
...
```

c. Run the following command on **ECS-target-01** to check whether it can receive packets:

tcpdump -i *<network-interface-name-of-the-mirror-target>* **udp port 4789 -nne**

Example command:

tcpdump -i eth0 udp port 4789 -nne

Information similar to the following is displayed:

```
[root@ecs-target-01 ~]# tcpdump -i eth0 udp port 4789 -nne
dropped privs to tcpdump
tcpdump: verbose output suppressed, use -v[v]... for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), snapshot length 262144 bytes
```

d. Enter any information (for example, **tcp**) on **ECS-test** and press **Enter** to send a TCP packet to **ECS-source**.

Information similar to the following is displayed:

```
[root@ecs-test ~]# nc 192.168.0.230 1234
hello
tcp
```

e. Check whether **ECS-source** can receive "tcp" from **ECS-test**.

If information similar to the following is displayed, **ECS-source** can receive "tcp" from **ECS-test**:

```
[root@ecs-source ~]# nc -l 1234
hello
tcp
```

f. Check whether **ECS-target-01** can receive the packet.

Information similar to the following is displayed. You can view the packet containing "tcp" sent by **ECS-test** after running **tcpdump**. **vni 1** is the identifier of **mirror-session-01**, indicating that **ECS-target-01** can receive the packet through this mirror session. The packet content has two parts: a VXLAN packet encapsulated by traffic mirroring and the original packet. For details, see **Table 13-27**.

```
[root@ecs-target-01 ~]# tcpdump -i eth0 udp port 4789 -nne
dropped privs to tcpdump
tcpdump: verbose output suppressed, use -v[v]... for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), snapshot length 262144 bytes
12:04:54.038631 fa:16:3e:d1:6b:5d > fa:16:3e:7e:d6:7a, ethertype IPv4 (0x0800), length 120:
192.168.0.230.32782 > 192.168.0.164.4789: VXLAN, flags [I] (0x08), vni 1
fa:16:3e:7e:d6:77 > fa:16:3e:7e:d6:bc, ethertype IPv4 (0x0800), length 70: 192.168.0.161.55602 >
192.168.0.230.1234: Flags [P.], seq 1838246001:1838246005, ack 2529760424, win 502, options
[nop,nop,TS val 1116821333 ecr 752395830], length 4
```

Step 5: Check Whether the UDP Mirror Session Works

1. Establish a UDP connection between **ECS-source** and **ECS-test**.

Send UDP packets from **ECS-test** to **ECS-source** and check whether **ECS-source** can receive the packets.

a. Run the following command on **ECS-source** to listen to port 1235:

nc -ul *<listening-port-of-mirror-source-ECS-source>*

Example command:

nc -ul 1235

If the command output is empty, the port is opened for listening.

b. Run the following command on **ECS-test** to establish a UDP connection between **ECS-source** and **ECS-test**:

nc *<private-IP-address-of-mirror-source-ECS-source> <listening-port-of-mirror-source-ECS-source>* **-u**

Example command:

nc 192.168.0.230 1235 -u

The command output is empty. Enter any information (for example, **hello**) on **ECS-test** and press **Enter** to check whether the UDP connection is successfully established.

```
[root@ecs-test ~]# nc 192.168.0.230 1235 -u
hello
```

c. Check whether **ECS-source** can receive information from **ECS-test**.

If information similar to the following is displayed, the UDP connection is successfully established.

```
[root@ecs-source ~]# nc -ul 1235
hello
```

2. Check whether the inbound UDP packets on **ECS-source** can be mirrored to **ECS-target-02**.

When **ECS-test** sends a UDP packet to **ECS-source**, run **tcpdump** to check whether **ECS-target-02** can receive the packet. If **ECS-target-02** receives the packet, the mirror session works.

a. Remotely log in to **ECS-target-02**.

For details, see **How Do I Log In to My ECS?**

b. Run the following command on **ECS-target-02** to view its network interface name:

ifconfig

Information similar to the following is displayed. In this example, the network interface of the mirror target is **eth0**.

```
[root@ecs-target-02 ~]# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.1.165  netmask 255.255.255.0  broadcast 192.168.1.255
        inet6 fe80::f816:3eff:fe7e:d77b  prefixlen 64  scopeid 0x20<link>
        ether fa:16:3e:7e:d7:7b  txqueuelen 1000  (Ethernet)
        RX packets 81142  bytes 112091279 (106.8 MiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 11848  bytes 2318498 (2.2 MiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
...
```

c. Run the following command on **ECS-target-02** to check whether it can receive packets:

tcpdump -i *<network-interface-name-of-the-mirror-target>* **udp port 4789 -nne**

Example command:

tcpdump -i eth0 udp port 4789 -nne

Information similar to the following is displayed:

```
[root@ecs-target-02 ~]# tcpdump -i eth0 udp port 4789 -nne
dropped privs to tcpdump
tcpdump: verbose output suppressed, use -v[v]... for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), snapshot length 262144 bytes
```

d. Enter any information (for example, **udp**) on **ECS-test** and press **Enter** to send a UDP packet to **ECS-source**.

Information similar to the following is displayed:

```
[root@ecs-test ~]# nc 192.168.0.230 1235 -u
hello
udp
```

e. Check whether **ECS-source** can receive information from **ECS-test**.

If information similar to the following is displayed, **ECS-source** can receive information from **ECS-test**:

```
[root@ecs-source ~]# nc -ul 1235
hello
udp
```

f. Check whether **ECS-target-02** can receive the packet.

Information similar to the following is displayed. You can view the packet containing **udp** sent by **ECS-test** after running **tcpdump**. **vni 2** is the identifier of **mirror-session-02**, indicating that **ECS-target-02** can receive the packet through this mirror session. The packet content has two parts: a VXLAN packet encapsulated by traffic mirroring and an original packet. For details, see **Table 13-27**.

```
[root@ecs-target-02 ~]# tcpdump -i eth0 udp port 4789 -nne
dropped privs to tcpdump
tcpdump: verbose output suppressed, use -v[v]... for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), snapshot length 262144 bytes
12:09:36.275574 fa:16:3e:18:32:b8 > fa:16:3e:7e:d7:7b, ethertype IPv4 (0x0800), length 96:
192.168.0.230.32830 > 192.168.1.165.4789: VXLAN, flags [I] (0x08), vni 2
fa:16:3e:7e:d6:77 > fa:16:3e:7e:d6:bc, ethertype IPv4 (0x0800), length 46: 192.168.0.161.46546 >
192.168.0.230.1235: UDP, length 4
```

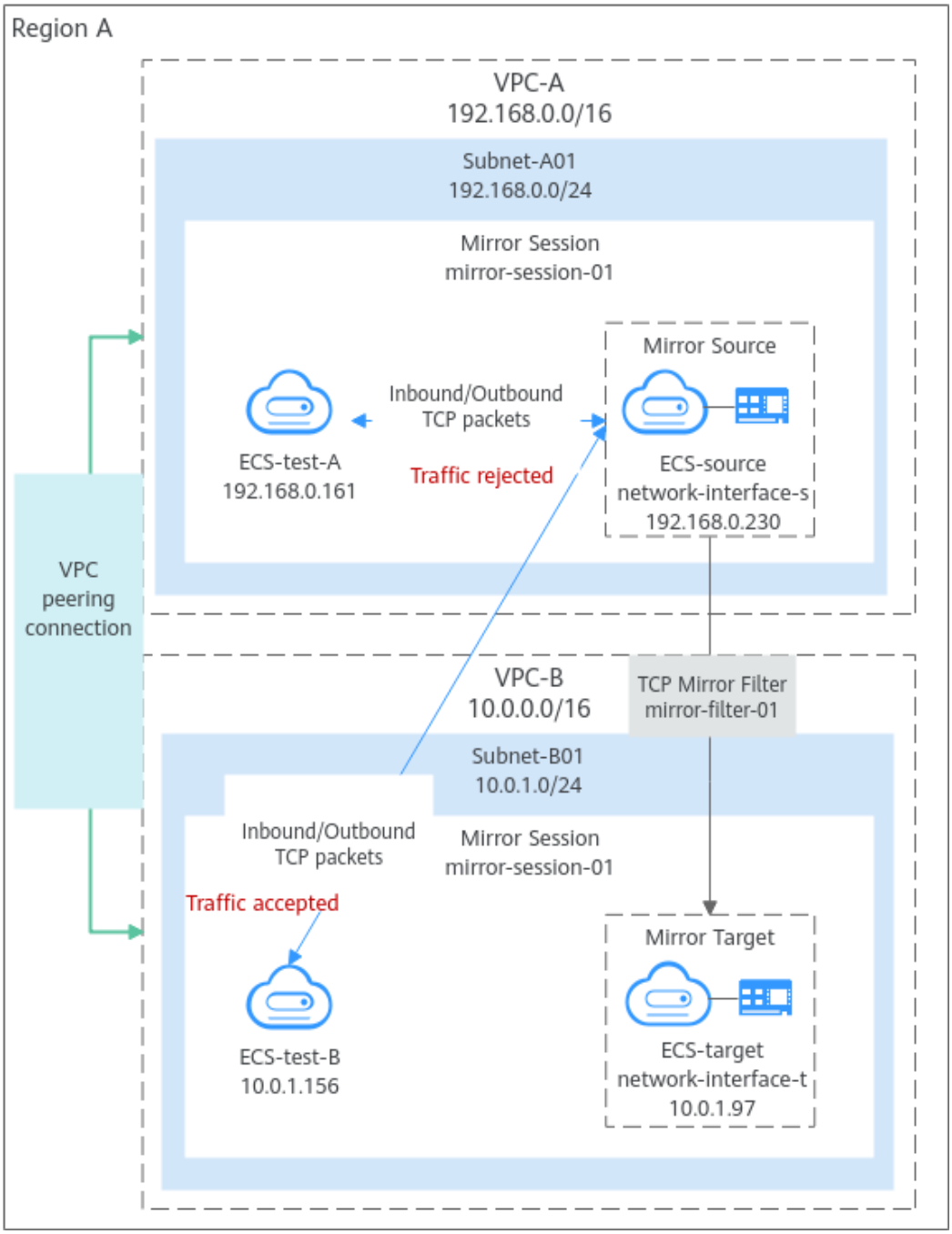
13.4.3 Mirroring Inbound and Outbound TCP Traffic to a Network Interface in a Different VPC

Solution Architecture

To mirror inbound and outbound TCP traffic between a mirror source (network interface) and a given instance to a mirror target (network interface) in a VPC different from the mirror source, you can refer to the configurations in this section. In **Figure 13-10**, mirror source **ECS-source** and mirror target **ECS-target** are running in different VPCs (**VPC-A** and **VPC-B** that are connected by a VPC peering connection). Traffic between **ECS-source** and **ECS-test-A** does not need to be mirrored. To mirror TCP traffic between **ECS-source** and **ECS-test-B** to **ECS-target**, a mirror session needs to be created. In this example, mirror session **mirror-session-01** is created. You can configure it as follows:

- Set the mirror source to **network-interface-s** on **ECS-source**. The inbound and outbound TCP traffic on this network interface will be mirrored.
- Set the mirror target to **network-interface-t** on **ECS-target**. The inbound and outbound TCP traffic on **network-interface-s** will be mirrored to **network-interface-t**.
- Create a mirror filter (**mirror-filter-01**) and add the following rules:
  - Two outbound rules: Rule 1 rejects TCP traffic from **ECS-source** to **ECS-test-A**. Rule 2 accepts TCP traffic from **ECS-source** to **ECS-test-B**.
  - Two inbound rules: Rule 1 rejects TCP traffic from **ECS-test-A** to **ECS-source**. Rule 2 accepts TCP traffic from **ECS-test-B** to **ECS-source**.

Figure 13-10 Mirroring inbound and outbound TCP traffic to a mirror target in a different VPC



Notes and Constraints

See **Notes and Constraints**.

Resource Planning

In this example, the VPCs, subnets, EIP, and ECSs must be in the same region but can be in different AZs.

NOTE

The following resource details are only for your reference. You can modify them if needed.

Table 13-33 Resource details for mirroring inbound and outbound TCP traffic

Resource	Quantity	Description
VPC and subnet	VPC: 2 Subnet: 2	Configure the VPCs as follows: • **Name**: Set it as needed. In this example, **VPC-A** and **VPC-B** are used. • **VPC IPv4 CIDR Block**: Set it as needed. In this example, **192.168.0.0/16** is used for **VPC-A**, and **10.0.0.0/16** is used for **VPC-B**. • **Subnet Name**: Set it as needed. In this example, there are two subnets: **Subnet-A01** in **VPC-A** and **Subnet-B01** in **VPC-B**. • **Subnet IPv4 CIDR Block**: Set it as needed. In this example, the CIDR block of **Subnet-A01** is **192.168.0.0/24** and that of **Subnet-B01** is **10.0.1.0/24**.

Resource	Quantity	Description
ECS	4	Configure the ECSs as follows: • **ECS Name**: Set it as needed. In this example, the ECSs are named **ECS-source**, **ECS-target**, **ECS-test-A**, and **ECS-test-B**. • **ECS Type**: In this example, the type of **ECS-source** is **General computing-plus c7t**. Currently, only network interfaces of ECSs of certain types can be used as mirror sources. For details, see **Notes and Constraints**. There are no constraints on the type of other ECSs. • **Image**: Set it as needed. In this example, public image **Huawei Cloud EulerOS 2.0 Standard 64 bit** is used. • **System Disk**: In this example, a general-purpose SSD disk of 40 GiB is used. • **Data Disk**: Set it as needed. In this example, no data disk is used. • **Network** – **VPC**: Select VPCs. In this example, select **VPC-A** for **ECS-source** and **ECS-test-A**, and **VPC-B** for **ECS-target** and **ECS-test-B**. – **Subnet**: Select subnets. In this example, select **Subnet-A01** for **ECS-source** and **ECS-test-A**, and **Subnet-B01** for **ECS-target** and **ECS-test-B**. • **Security Group**: In this example, the four ECSs are associated with the same security group (**Sg-X**). Ensure that all rules in **Table 13-34** are added. If the ECSs are associated with different security groups, you also need to add additional rules. – If **ECS-test-A** is associated with **Sg-X** but **ECS-source** is associated with **Sg-A**, add the rules in **Table 13-35** to **Sg-A** and **Sg-X** to allow traffic between **ECS-test-A** and **ECS-source**. The same applies to **ECS-test-B**. – If **ECS-source** is associated with **Sg-A** but **ECS-target** is associated with **Sg-B**, add the rules in **Table 13-36** to **Sg-B** to allow UDP packets encapsulated by the mirror source to access the mirror target over port 4789. • EIP: Select **Not required**. • Private IP address: In this example, use **192.168.0.230** for **ECS-source**, **10.0.1.97** for **ECS-target**, **192.168.0.161** for **ECS-test-A**, and **10.0.1.156** for **ECS-test-B**.

Resource	Quantity	Description
EIP	1	• **Billing Mode**: Set it as needed. In this example, **Pay-per-use** is used. • **EIP Name**: Set it as needed. In this example, **EIP-A** is used. • EIP: The EIP is randomly assigned. In this example, **124.X.X.187** is used.
VPC peering connection	1	• **VPC Peering Connection Name**: Set it as needed. In this example, **Peering-AB** is used. • **Local VPC**: Select a VPC as needed. In this example, select **VPC-A** and its CIDR block is **192.168.0.0/16**. • **Account**: In this example, **VPC-A** and **VPC-B** are in the same account. Select **My account**. Traffic cannot be mirrored across VPCs in different accounts. • **Peer Project**: Retain the default value. • **Peer VPC**: Select a VPC as needed. In this example, select **VPC-B** and its CIDR block is **10.0.0.0/16**. • Add the routes in **Table 13-37** for the VPC peering connection.
Mirror filter	1	• **Name**: Set it as needed. In this example, **mirror-filter-01** is used. • Inbound rules: Add the two inbound rules in **Table 13-38**. – Rule 1: rejects TCP traffic from all instances, including **ECS-test-A**, in **VPC-A** to mirror source **ECS-source**. – Rule 2: accepts TCP traffic from all instances, including **ECS-test-B**, in **VPC-B** to mirror source **ECS-source**. • Outbound rules: Add the two outbound rules in **Table 13-38**. – Rule 1: rejects TCP traffic from mirror source **ECS-source** to instances, including **ECS-test-A**, in **VPC-A**. – Rule 2: accepts TCP traffic from mirror source **ECS-source** to instances, including **ECS-test-B**, in **VPC-B**.

Resource	Quantity	Description
Mirror session	1	• **Basic Information** – **Name**: Set it as needed. In this example, **mirror-session-01** is used. – **Priority**: Set it as needed. In this example, **1** is used. – **VNI**: Set it as needed. In this example, **1** is used. – **Packet Length**: Set it as needed. In this example, **96** is used. – **Mirror Session**: Enable it to mirror the traffic from the mirror source. • **Associate Mirror Filter**: Set it as needed. In this example, **mirror-filter-01** is used. • **Associate Mirror Sources**: Set it as needed. In this example, the private IP address (192.168.0.230) of the network interface on **ECS-source** is used. • **Associate Mirror Target** – **Type**: **Network interface** – Network interface: Set it as needed. In this example, the private IP address (10.0.1.97) of the network interface of **ECS-target** is used.

Table 13-34 Security group **Sg-X** rules

Direction	Action	Type	Protocol & Port	Source/ Destination	Description
Inbound	Allow	IPv4	TCP: 22	Source: 0.0.0.0/0	Allows remote logins to Linux ECSs over SSH port 22.
Inbound	Allow	IPv4	TCP: 3389	Source: 0.0.0.0/0	Allows remote logins to Windows ECSs over RDP port 3389.
Inbound	Allow	IPv4	All	Source: current security group (**Sg-X**)	Allows the ECSs in this security group to communicate with each other using IPv4 addresses.
Inbound	Allow	IPv6	All	Source: current security group (**Sg-X**)	Allows the ECSs in this security group to communicate with each other using IPv6 addresses.

Direction	Action	Type	Protocol & Port	Source/ Destination	Description
Outbound	Allow	IPv4	All	Destination: 0.0.0.0/0	Allows ECSs in this security group to access the Internet using IPv4 addresses.
Outbound	Allow	IPv6	All	Destination: ::/0	Allows ECSs in this security group to access the Internet using IPv6 addresses.

NOTICE

If the source of an inbound rule is set to 0.0.0.0/0, all external IP addresses are allowed to remotely log in to your cloud server. Exposing port 22 or 3389 to the public network will leave your instances vulnerable to network risks. To address this issue, set the source to a trusted IP address, for example, the IP address of your local PC.

Table 13-35 Rules for security groups **Sg-A** and **Sg-X** to allow traffic between ECSs

Security Group	Direction	Action	Type	Protocol & Port	Source	Description
Sg-A	Inbound	Allow	IPv4	TCP: 1234	The security group with which **ECS-test-A** is associated: **Sg-X**	Allows TCP packets from **ECS-test-A** to **ECS-source** over port 1234.
Sg-X	Inbound	Allow	IPv4	TCP: All ports	The security group with which **ECS-source** is associated: **Sg-A**	Allows TCP packets from **ECS-source** to **ECS-test-A** over all ports.

Table 13-36 Security group **Sg-B** rule

Direction	Action	Type	Protocol & Port	Source	Description
Inbound	Allow	IPv4	UDP: 4789	The private IP address of mirror source **ECS-source**: 192.168.0.230/32	Allows UDP packets encapsulated by **ECS-source** to access **ECS-target** over port 4789.

Table 13-37 Routes for the VPC peering connection

VPC	Route Table	Destination	Next Hop	Description
VPC-A	rtb-VPC-A (default)	VPC-B CIDR block: 10.0.0.0/16	VPC peering connection: **Peering-AB**	Route from **VPC-A** to **VPC-B**
VPC-B	rtb-VPC-B (default)	VPC-A CIDR block: 192.168.0.0/16	VPC peering connection: **Peering-AB**	Route from **VPC-B** to **VPC-A**

Table 13-38 Inbound and outbound rules of the mirror filter

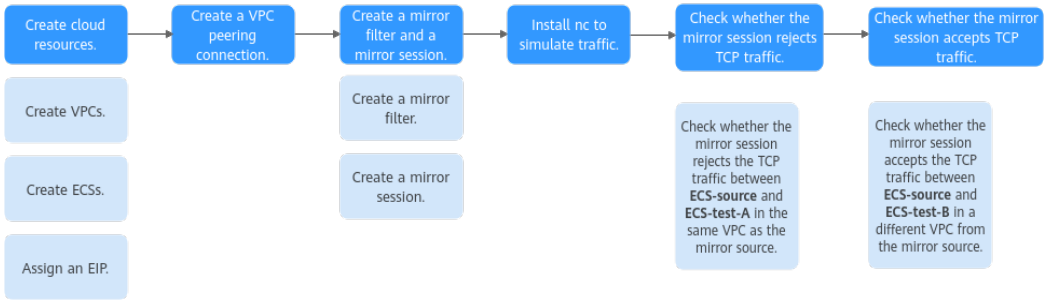
Direction	Priority	Protocol	Action	Type	Source	Source Port Range	Destination	Destination Port Range
Inbound	1	TCP	Reject	IPv4	**VPC-A** CIDR block: 192.168.0.0/16	All	**VPC-A** CIDR block: 192.168.0.0/16	All
Inbound	2	TCP	Accept	IPv4	**VPC-B** CIDR block: 10.0.0.0/16	All	The private IP address of **ECS-source**: 192.168.0.230/32	Port of **ECS-source**: 1234-1234
Outbound	1	TCP	Reject	IPv4	**VPC-A** CIDR block: 192.168.0.0/16	All	**VPC-A** CIDR block: 192.168.0.0/16	All

Direction	Priority	Protocol	Action	Type	Source	Source Port Range	Destination	Destination Port Range
Outbound	2	TCP	Accept	IPv4	The private IP address of **ECS-source**: 192.168.0.230/32	All	**VPC-B** CIDR block: 10.0.0.0/16	Port of **ECS-test-B**: 1234-1234

Procedure

Figure 13-11 shows the procedure required to mirror the inbound and outbound TCP traffic between a mirror source (network interface) and a given instance to a mirror target (network interface) in a different VPC from the mirror source.

Figure 13-11 Mirroring inbound and outbound TCP traffic to a mirror target in a different VPC



Step 1: Create Cloud Resources

1. Create two VPCs, each with a subnet.

   For details, see **Creating a VPC and Subnet**.
2. Create four ECSs.

   For details, see **Purchasing a Custom ECS**.
3. Assign an EIP.

   For details, see **Assigning an EIP**.

Step 2: Create a VPC Peering Connection

Create a VPC peering connection to connect **VPC-A** and **VPC-B** by referring to **Creating a VPC Peering Connection to Connect Two VPCs in the Same Account**.

Add forward and return routes to the route tables of **VPC-A** and **VPC-B** so that the two VPCs can communicate with each other. For details, see **Table 13-37**.

Step 3: Create a Mirror Filter and a Mirror Session

1. Create a mirror filter.

   For details, see **Creating a Mirror Filter**.

2. Create a mirror session, and associate the mirror filter, mirror source, and mirror target with this mirror session.

   For details, see **Creating a Mirror Session**.

Step 4: Install Netcat (nc) to Simulate Traffic

The nc utility reads and writes data across network connections using TCP or UDP. It is usually used to test ports for accessibility. You need to install nc on **ECS-source**, **ECS-test-A**, and **ECS-test-B**.

1. Install nc on **ECS-source**.

   a. Bind the EIP to **ECS-source** to connect to the Internet for downloading the nc utility.

      For details, see **Binding an EIP to an ECS**.

   b. Remotely log in to **ECS-source**.

      For details, see **How Do I Log In to My ECS?**

   c. Run the following commands in sequence to install nc:

      sudo yum update

      Information similar to the following is displayed:

```
[root@ecs-source ~]# sudo yum update
HCE 2.0
base
         55 MB/s | 6.1 MB     00:00
HCE 2.0
updates
           98 MB/s |  14 MB     00:00
Last metadata expiration check: 0:00:01 ago on Tue 10 Sep 2024 05:54:28 PM CST.
Dependencies resolved.
Nothing to do.
Complete!
```

      sudo yum install nc

      If information similar to the following is displayed, enter **y** as prompted and press **Enter**:

```
[root@ecs-source ~]# sudo yum install nc
Last metadata expiration check: 0:00:12 ago on Tue 10 Sep 2024 05:54:28 PM CST.
Dependencies resolved.
...
Install  2 Packages

Total download size: 6.1 M
Installed size: 25 M
Is this ok [y/N]: y
Downloading Packages:
...
Importing GPG key 0xA8DEF926:
 Userid     : "HCE <support@huaweicloud.com>"
 Fingerprint: C1BA 9CD4 9D03 A206 E241 F176 28DA 5B77 A8DE F926
 From       : http://repo.huaweicloud.com/hce/2.0/updates/RPM-GPG-KEY-HCE-2
Is this ok [y/N]: y
...
Installed:
  libssh2-1.10.0-2.r10.hce2.x86_64
nmap-2:7.92-2.r4.hce2.x86_64
```



```
Complete!
```

d. Unbind the EIP from **ECS-source** after nc is installed.

For details, see **Unbinding an EIP**.

2. Repeat **1.a** to **1.d** on **ECS-test-A**.
3. Repeat **1.a** to **1.d** on **ECS-test-B**.
4. Release the EIP.

For details, see **Unbinding an EIP**. If you do not release the EIP, the EIP will continue to be billed.

Step 5: Check Whether the Mirror Session Rejects the Traffic Between ECS-source and ECS-test-A

Check whether the mirror session rejects the traffic between **ECS-source** and **ECS-test-A**.

1. Establish a TCP connection between **ECS-source** and **ECS-test-A**.

Use **ECS-source** to send TCP packets to **ECS-test-A** and check whether **ECS-test-A** can receive the packets.

a. Run the following command on **ECS-source** to listen to port 1234:

nc -l *<listening-port-of-mirror-source-ECS-source>*

Example command:

nc -l 1234

If the command output is empty, the port is opened for listening.

b. Run the following command on **ECS-test-A** to establish a TCP connection between **ECS-source** and **ECS-test-A**:

nc *<private-IP-address-of-mirror-source-ECS-source> <listening-port-of-mirror-source-ECS-source>*

Example command:

nc 192.168.0.230 1234

If the command output is empty, the TCP connection has been established.

c. Enter any information (for example, **hello**) on **ECS-source** and press **Enter** to check whether requests can be sent over the TCP connection.

```
[root@ecs-source ~]# nc -l 1234
hello
```

d. Check whether **ECS-test-A** can receive "hello" from **ECS-source**.

If information similar to the following is displayed, **ECS-test-A** receives "hello" from **ECS-source**.

```
[root@ecs-test-a ~]# nc 192.168.0.230 1234
hello
```

2. Check whether the outbound TCP packet from **ECS-source** to **ECS-test-A** can be mirrored to **ECS-target**.

When **ECS-source** sends a TCP packet to **ECS-test-A**, run **tcpdump** to check whether **ECS-target** can receive the packet. If **ECS-target** does not receive the packet, the mirror session rejects the outbound TCP traffic.

a. Remotely log in to **ECS-target**.

For details, see **How Do I Log In to My ECS?**

b. Run the following command on **ECS-target** to view its network interface name:

ifconfig

Information similar to the following is displayed. In this example, the network interface of the mirror target is **eth0**.

```
[root@ecs-target ~]# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.0.1.97  netmask 255.255.255.0  broadcast 10.0.1.255
        inet6 fe80::f816:3eff:fea0:a101  prefixlen 64  scopeid 0x20<link>
        ether fa:16:3e:a0:a1:01  txqueuelen 1000  (Ethernet)
        RX packets 103445  bytes 119352826 (113.8 MiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 34118  bytes 15630293 (14.9 MiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
...
```

c. Run the following command on **ECS-target** to check whether it can receive the packet:

tcpdump -i *<network-interface-name-of-the-mirror-target>* **udp port 4789 -nne**

Example command:

tcpdump -i eth0 udp port 4789 -nne

Information similar to the following is displayed:

```
[root@ecs-target ~]# tcpdump -i eth0 udp port 4789 -nne
dropped privs to tcpdump
tcpdump: verbose output suppressed, use -v[v]... for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), snapshot length 262144 bytes
```

d. Enter any information (for example, **to testa**) on **ECS-source** and press **Enter** to send a TCP packet to **ECS-test-A**.

Information similar to the following is displayed:

```
[root@ecs-source ~]# nc -l 1234
hello
to testa
```

e. Check whether **ECS-test-A** can receive "to testa" from **ECS-source**.

If information similar to the following is displayed, **ECS-test-A** can receive "to testa" from **ECS-source**.

```
[root@ecs-test-a ~]# nc 192.168.0.230 1234
hello
to testa
```

f. Check whether **ECS-target** can receive the packet.

If the information similar to the following is displayed, the packet containing "to testa" from **ECS-source** is not sent to **ECS-test-A** after running **tcpdump**. This means the reject rule works.

```
[root@ecs-target ~]# tcpdump -i eth0 udp port 4789 -nne
dropped privs to tcpdump
tcpdump: verbose output suppressed, use -v[v]... for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), snapshot length 262144 bytes
```

3. Check whether the inbound TCP packets from **ECS-test-A** to **ECS-source** can be mirrored to **ECS-target**.

When **ECS-test-A** sends a TCP packet to **ECS-source**, run **tcpdump** to check whether **ECS-target** can receive the packet. If **ECS-target** does not receive the packet, the mirror session rejects the inbound TCP traffic.

a. Enter any information (for example, **testa to source**) on **ECS-test-A** and press **Enter** to send a TCP packet to **ECS-source**.

Information similar to the following is displayed:

```
[root@ecs-test-a ~]# nc 192.168.0.230 1234
hello
to testa
testa to source
```

b. Check whether **ECS-source** can receive "testa to source" from **ECS-test-A**.

If information similar to the following is displayed, **ECS-source** can receive "testa to source" from **ECS-test-A**.

```
[root@ecs-source ~]# nc -l 1234
hello
to testa
testa to source
```

c. Check whether **ECS-target** can receive the TCP packet.

If the information similar to the following is displayed, the packet containing "testa to source" from **ECS-test-A** is not sent to **ECS-source** after running **tcpdump**. This means the reject rule works.

```
[root@ecs-target ~]# tcpdump -i eth0 udp port 4789 -nne
dropped privs to tcpdump
tcpdump: verbose output suppressed, use -v[v]... for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), snapshot length 262144 bytes
```

Step 6: Check Whether the Mirror Session Accepts the Traffic Between ECS-source and ECS-test-B

Check whether the mirror session accepts the traffic between **ECS-source** and **ECS-test-B**.

1. Establish a TCP connection between **ECS-source** and **ECS-test-B**.

   Use **ECS-test-B** to send TCP packets to **ECS-source** and check whether **ECS-source** can receive the packets.

   a. Run the following command on **ECS-test-B** to listen to port 1234:

   nc -l *<listening-port-of-ECS-test-B>*

   Example command:

   nc -l 1234

   If the command output is empty, the port is opened for listening.

   b. Run the following command on **ECS-source** to establish a TCP connection between **ECS-source** and **ECS-test-B**:

   nc *<private-IP-address-of-ECS-test-B> <listening-port-of-ECS-test-B>*

   Example command:

   nc 10.0.1.156 1234

   If the command output is empty, the TCP connection has been established.

   c. Enter any information (for example, **hello**) on **ECS-test-B** and press **Enter** to check whether requests can be sent over the TCP connection.

```
[root@ecs-test-b ~]# nc -l 1234
hello
```

   d. Check whether **ECS-source** can receive "hello" from **ECS-test-B**.

   If information similar to the following is displayed, **ECS-source** can receive "hello" from **ECS-test-B**.

```
[root@ecs-source ~]# nc 10.0.1.156 1234
hello
```


2. Check whether the outbound TCP packet from **ECS-source** to **ECS-test-B** can be mirrored to **ECS-target**.

   When **ECS-source** sends a TCP packet to **ECS-test-B**, run **tcpdump** to check whether **ECS-target** can receive the packet. If **ECS-target** receives the packet, the mirror session accepts the outbound TCP traffic.

   a. Remotely log in to **ECS-target**.

      For details, see **How Do I Log In to My ECS?**

   b. Run the following command on **ECS-target** to view its network interface name:

      ifconfig

      Information similar to the following is displayed. In this example, the network interface of the mirror target is **eth0**.

      ```
      [root@ecs-target ~]# ifconfig
      eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
            inet 10.0.1.97  netmask 255.255.255.0  broadcast 10.0.1.255
            inet6 fe80::f816:3eff:fea0:a101  prefixlen 64  scopeid 0x20<link>
            ether fa:16:3e:a0:a1:01  txqueuelen 1000  (Ethernet)
            RX packets 103445  bytes 119352826 (113.8 MiB)
            RX errors 0  dropped 0  overruns 0  frame 0
            TX packets 34118  bytes 15630293 (14.9 MiB)
            TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
      ...
      ```

   c. Run the following command on **ECS-target** to check whether it can receive packets:

      tcpdump -i *<network-interface-name-of-the-mirror-target>* **udp port 4789 -nne**

      Example command:

      tcpdump -i eth0 udp port 4789 -nne

      Information similar to the following is displayed:

      ```
      [root@ecs-target ~]# tcpdump -i eth0 udp port 4789 -nne
      dropped privs to tcpdump
      tcpdump: verbose output suppressed, use -v[v]... for full protocol decode
      listening on eth0, link-type EN10MB (Ethernet), snapshot length 262144 bytes
      ```

   d. Enter any information (for example, **to testb**) on **ECS-source** and press **Enter** to send a TCP packet to **ECS-test-B**.

      Information similar to the following is displayed:

      ```
      [root@ecs-source ~]# nc 10.0.1.156 1234
      hello
      to testb
      ```

   e. Check whether **ECS-test-B** can receive "to testb" from **ECS-source**.

      If information similar to the following is displayed, **ECS-test-B** can receive "to testb" from **ECS-source**.

      ```
      [root@ecs-test-b ~]# nc -l 1234
      hello
      to testb
      ```

   f. Check whether **ECS-target** can receive the TCP packet.

      If the information similar to the following is displayed, the packet containing "to testb" (time: **17:28:48.772658**) from **ECS-source** is sent to **ECS-test-B** after running **tcpdump**. This means the accept rule works. In this packet, **vni 1** is the identifier of **mirror-session-01**, indicating that **ECS-target** can receive the packet through this mirror session. The packet content has two parts: a VXLAN packet encapsulated by traffic mirroring and the original packet. For details, see **Table 13-27**.


```
[root@ecs-target ~]# tcpdump -i eth0 udp port 4789 -nne
dropped privs to tcpdump
tcpdump: verbose output suppressed, use -v[v]... for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), snapshot length 262144 bytes
17:28:48.772658 fa:16:3e:6e:42:80 > fa:16:3e:a0:a1:01, ethertype IPv4 (0x0800), length 125:
192.168.0.230.32821 > 10.0.1.97.4789: VXLAN, flags [I] (0x08), vni 1
fa:16:3e:7e:d6:bc > fa:16:3e:d1:6b:5d, ethertype IPv4 (0x0800), length 75: 192.168.0.230.44906 >
10.0.1.156.1234: Flags [P.], seq 935460393:935460402, ack 4279496885, win 502, options
[nop,nop,TS val 1414482596 ecr 3323401462], length 9
```

3. Check whether the inbound TCP packets from **ECS-test-B** to **ECS-source** can be mirrored to **ECS-target**.

   When **ECS-test-B** sends a TCP packet to **ECS-source**, run **tcpdump** to check whether **ECS-target** can receive the packet. If **ECS-target** receives the packet, the mirror session accepts the inbound TCP traffic.

   a. Enter any information (for example, **testb to source**) on **ECS-test-B** and press **Enter** to send a TCP packet to **ECS-source**.

      Information similar to the following is displayed:

      ```
      [root@ecs-test-b ~]# nc -l 1234
      hello
      to testb
      testb to source
      ```

   b. Check whether **ECS-source** can receive "testb to source" from **ECS-test-B**.

      If information similar to the following is displayed, **ECS-source** can receive "testb to source" from **ECS-test-B**.

      ```
      [root@ecs-source ~]# nc 10.0.1.156 1234
      hello
      to testb
      testb to source
      ```

   c. Check whether **ECS-target** can receive the TCP packet.

      If the information similar to the following is displayed, the packet containing "testb to source" (time: **17:30:26.193420**) from **ECS-test-B** is sent to **ECS-source** after running **tcpdump**. This means the accept rule works. In this packet, **vni 1** is the identifier of **mirror-session-01**, indicating that **ECS-target** can receive the packet through this mirror session. The packet content has two parts: a VXLAN packet encapsulated by traffic mirroring and the original packet. For details, see **Table 13-27**.

      ```
      [root@ecs-target ~]# tcpdump -i eth0 udp port 4789 -nne
      dropped privs to tcpdump
      tcpdump: verbose output suppressed, use -v[v]... for full protocol decode
      listening on eth0, link-type EN10MB (Ethernet), snapshot length 262144 bytes
      17:28:48.772658 fa:16:3e:6e:42:80 > fa:16:3e:a0:a1:01, ethertype IPv4 (0x0800), length 125:
      192.168.0.230.32821 > 10.0.1.97.4789: VXLAN, flags [I] (0x08), vni 1
      fa:16:3e:7e:d6:bc > fa:16:3e:d1:6b:5d, ethertype IPv4 (0x0800), length 75: 192.168.0.230.44906 >
      10.0.1.156.1234: Flags [P.], seq 935460393:935460402, ack 4279496885, win 502, options
      [nop,nop,TS val 1414482596 ecr 3323401462], length 9
      17:30:26.193420 fa:16:3e:6e:42:80 > fa:16:3e:a0:a1:01, ethertype IPv4 (0x0800), length 116:
      192.168.0.230.32821 > 10.0.1.97.4789: VXLAN, flags [I] (0x08), vni 1
      fa:16:3e:7e:d6:bc > fa:16:3e:d1:6b:5d, ethertype IPv4 (0x0800), length 66: 192.168.0.230.44906 >
      10.0.1.156.1234: Flags [.], ack 17, win 502, options [nop,nop,TS val 1414580016 ecr
      3323563970], length 0
      ```

13.4.4 Mirroring Inbound and Outbound TCP Traffic to a Load Balancer

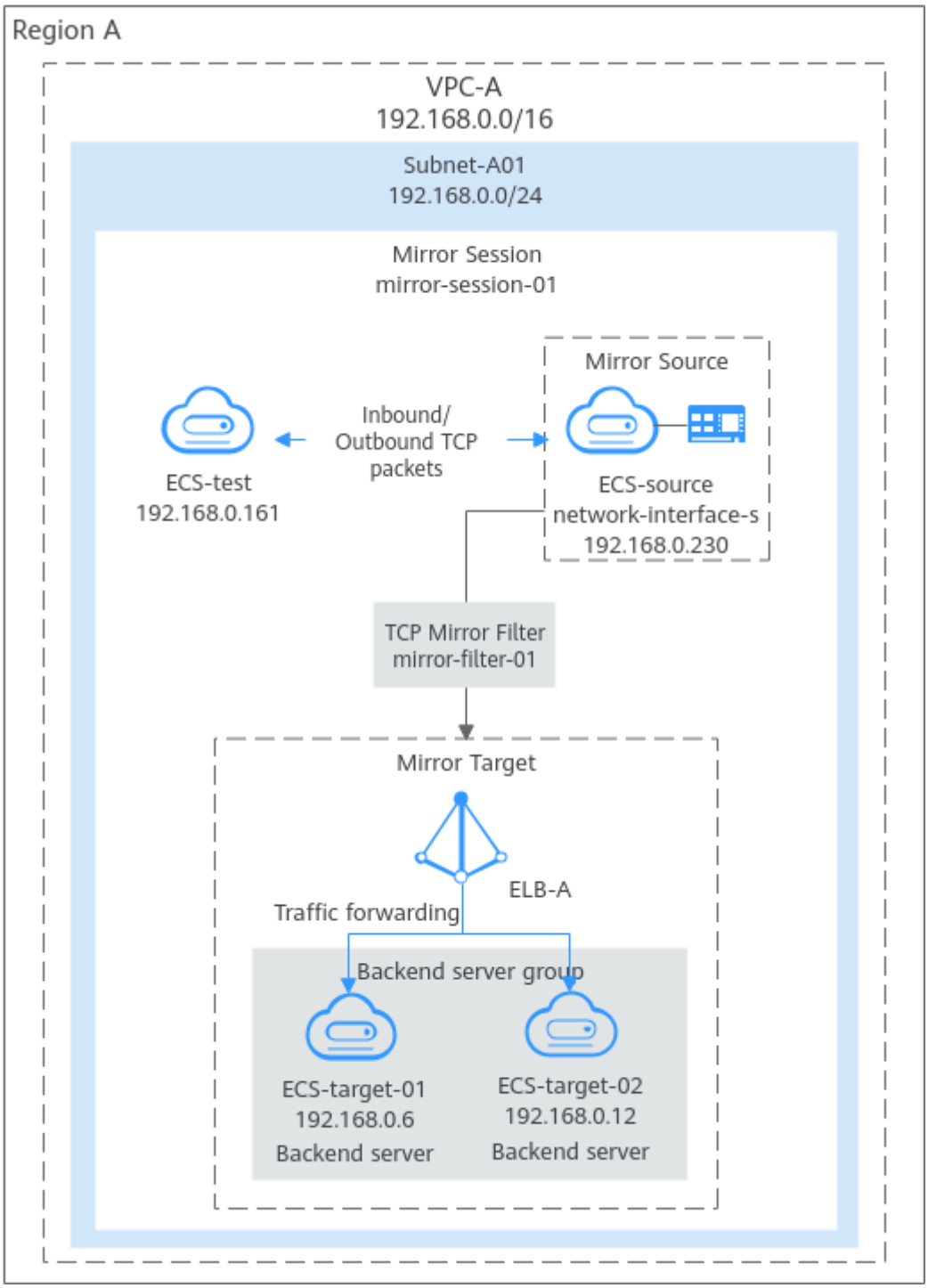
Solution Architecture

To mirror inbound and outbound TCP traffic between a mirror source (network interface) and a given instance to a mirror target (load balancer), you can refer to

the configurations in this section. In **Figure 13-12**, to mirror the TCP traffic between **ECS-source** and **ECS-test** running in VPC **VPC-A** to load balancer **ELB-A**, one mirror session is needed. In this example, mirror session **mirror-session-01** is created. You can configure it as follows:

- Set the mirror source to **network-interface-s** on **ECS-source**. The inbound and outbound TCP traffic on this network interface will be mirrored.
- Set the mirror target to **ELB-A**. The inbound and outbound TCP traffic on **network-interface-s** will be mirrored to **ELB-A**. **ELB-A** will distribute the mirrored traffic across backend servers **ECS-target-01** and **ECS-target-02** based on the routing rules.
- Create a mirror filter (**mirror-filter-01**) and add the following rules:
  - Outbound rule: accepts TCP traffic from **ECS-source** to **ECS-test**.
  - Inbound rule: accepts TCP traffic from **ECS-test** to **ECS-source**.

Figure 13-12 Mirroring inbound and outbound TCP traffic to a load balancer



Notes and Constraints

See **Notes and Constraints**.

Resource Planning

In this example, the VPC, subnet, EIP, load balancer, and ECSs must be in the same region but can be in different AZs.

NOTE

The following resource details are only for your reference. You can modify them if needed.

Table 13-39 Resource details for mirroring inbound and outbound TCP traffic to a load balancer

Resource	Quantity	Description
VPC and subnet	VPC: 1 Subnet: 1	• **Name**: Set it as needed. In this example, **VPC-A** is used. • **VPC IPv4 CIDR Block**: Set it as needed. In this example, **192.168.0.0/16** is used. • **Subnet Name**: Set it as needed. In this example, **Subnet-A01** is used. • **Subnet IPv4 CIDR Block**: Set it as needed. In this example, **192.168.0.0/24** is used.

Resource	Quantity	Description
ECS	4	Configure the ECSs as follows: • **ECS Name**: Set it as needed. In this example, the ECSs are named **ECS-source**, **ECS-target-01**, **ECS-target-02**, and **ECS-test**. • **ECS Type**: In this example, the type of **ECS-source** is **General computing-plus c7t**. Currently, only network interfaces of ECSs of certain types can be used as mirror sources. For details, see **Notes and Constraints**. There are no constraints on the type of other ECSs. • **Image**: Set it as needed. In this example, public image **Huawei Cloud EulerOS 2.0 Standard 64 bit** is used. • **System Disk**: In this example, a general-purpose SSD disk of 40 GiB is used. • **Data Disk**: Set it as needed. In this example, no data disk is used. • **Network** – **VPC**: Select a VPC. In this example, **VPC-A** is used. – **Subnet**: Select a subnet. In this example, **Subnet-A01** is used. • **Security Group**: In this example, the four ECSs are associated with the same security group (**Sg-X**). Ensure that all rules in **Table 13-40** are added. If the ECSs are associated with different security groups, you also need to add additional rules. – If **ECS-test** is associated with **Sg-X** but **ECS-source** is associated with **Sg-A**, add the rules in **Table 13-41** to **Sg-A** and **Sg-X** to allow traffic between **ECS-test** and **ECS-source**. – If **ECS-source** is associated with **Sg-A** and **ECS-target-01** is associated with **Sg-B**, add the rule in **Table 13-42** to **Sg-B** to allow UDP packets encapsulated by the mirror source to access the mirror target over port 4789. The same applies to **ECS-target-02**. • **EIP**: Select **Not required**. • Private IP address: In this example, use **192.168.0.230** for **ECS-source**, **192.168.0.6** for **ECS-target-01**, **192.168.0.12** for **ECS-target-02**, and **192.168.0.161** for **ECS-test**.

Resource	Quantity	Description
EIP	1	• **Billing Mode**: Set it as needed. In this example, **Pay-per-use** is used. • **EIP Name**: Set it as needed. In this example, **EIP-A** is used. • EIP: The EIP is randomly assigned. In this example, **124.***X.X.***187** is used.
Load balancer	1	• Target type: Only dedicated load balancers can be used as mirror targets. • **Name**: Set it as needed. In this example, **ELB-A** is used. • **Specifications**: Select **Network load balancing (TCP/UDP/TLS)** to forward UDP packets. Specifications vary by region. If TLS is not supported in your region, select **Network load balancing (TCP/UDP)**. • **Network Type**: The encapsulated mirrored packet uses the IPv4 UDP protocol. Select **Private IPv4 network** in this example. • **VPC**: Set it as needed. In this example, select **VPC-A**. • **Frontend Subnet**: Set it as needed. In this example, select **Subnet-A01**. • IPv4 address: Set it as needed. In this example, select **Automatically assign IP address**. • **Backend Subnet**: Set it as needed. In this example, select **Subnet of the load balancer**. • **IP as a Backend**: Disable it. • **EIP**: Set it as needed. In this example, select **Not required**.

Resource	Quantity	Description
Listener	1	• Configure the listener as follows: – **Name**: Set it as needed. In this example, name it **listener-A**. – **Frontend Protocol**: Select **UDP** to forward encapsulated UDP packets. – **Frontend Port**: Enter **4789**, the fixed port used by the mirror target to receive traffic. – **Access Control**: Select **All IP addresses** for this example. If you select **Whitelist**, ensure that the mirror source IP address is in the whitelist. • Configure the routing policy as follows: – **Forwarding Mode**: Set it as needed. In this example, select **Load balancing**. – **Backend Server Group Type**: Set it as needed. In this example, select **Hybrid**. – **Backend Server Group Name**: Set it as needed. In this example, name it **serve_group-A**. – **Backend Protocol**: Select **UDP** to process encapsulated UDP packets. – **Forward to Same Port**: Set it as needed. In this example, disable it. – **Load Balancing Algorithm**: Set it as needed. In this example, select **Weighted round robin**. – **Sticky Session**: Set it as needed. In this example, disable it. • Add backend servers as follows: – On the **Backend Server** tab, select **ECS-target-01** and **ECS-target-02** for this example. Set **Backend Port** to **4789** and **Weight** to **1**. – **Health Check**: Disable it for this example. If port 4789 is not listened to and the health check option is enabled, the health check result will be unhealthy, preventing ELB from forwarding traffic to backend servers.
Mirror filter	1	• **Name**: Set it as needed. In this example, **mirror-filter-01** is used. • Inbound rule: Add an inbound rule to accept TCP traffic from **ECS-test** to **ECS-source**. • Outbound rule: Add an outbound rule to accept TCP traffic from **ECS-source** to **ECS-test**. For details about the rules, see **Table 13-43**.

Resource	Quantity	Description
Mirror session	1	• **Basic Information** – **Name**: Set it as needed. In this example, **mirror-session-01** is used. – **Priority**: Set it as needed. In this example, **1** is used. – **VNI**: Set it as needed. In this example, **1** is used. – **Packet Length**: Set it as needed. In this example, **96** is used. – **Mirror Session**: Enable it to mirror the traffic from the mirror source. • **Associate Mirror Filter**: Set it as needed. In this example, **mirror-filter-01** is used. • **Associate Mirror Sources**: Set it as needed. In this example, the private IP address (192.168.0.230) of the network interface of **ECS-source** is used. • **Associate Mirror Target** – **Type**: **Load balancer** – Network interface: Set it as needed. In this example, the private IP address (192.168.0.147) of the network interface of **ELB-A** is used.

Table 13-40 Security group **Sg-X** rules

Direction	Action	Type	Protocol & Port	Source/ Destination	Description
Inbound	Allow	IPv4	TCP: 22	Source: 0.0.0.0/0	Allows remote logins to Linux ECSs over SSH port 22.
Inbound	Allow	IPv4	TCP: 3389	Source: 0.0.0.0/0	Allows remote logins to Windows ECSs over RDP port 3389.
Inbound	Allow	IPv4	All	Source: current security group (**Sg-X**)	Allows the ECSs in this security group to communicate with each other using IPv4 addresses.
Inbound	Allow	IPv6	All	Source: current security group (**Sg-X**)	Allows the ECSs in this security group to communicate with each other using IPv6 addresses.

Direction	Action	Type	Protocol & Port	Source/ Destination	Description
Outbound	Allow	IPv4	All	Destination: 0.0.0.0/0	Allows ECSs in this security group to access the Internet using IPv4 addresses.
Outbound	Allow	IPv6	All	Destination: ::/0	Allows ECSs in this security group to access the Internet using IPv6 addresses.

NOTICE

If the source of an inbound rule is set to 0.0.0.0/0, all external IP addresses are allowed to remotely log in to your cloud server. Exposing port 22 or 3389 to the public network will leave your instances vulnerable to network risks. To address this issue, set the source to a trusted IP address, for example, the IP address of your local PC.

Table 13-41 Rules of security groups **Sg-A** and **Sg-X** (allowing traffic between ECSs)

Security Group	Direction	Action	Type	Protocol & Port	Source	Description
Sg-A	Inbound	Allow	IPv4	TCP: 1234	The security group with which **ECS-test** is associated: **Sg-X**	Allows TCP packets from **ECS-test** to **ECS-source** over port 1234.
Sg-X	Inbound	Allow	IPv4	TCP: All ports	The security group with which **ECS-source** is associated: **Sg-A**	Allows TCP packets from **ECS-source** to **ECS-test** over all ports.

Table 13-42 Security group **Sg-B** rule

Direction	Action	Type	Protocol & Port	Source	Description
Inbound	Allow	IPv4	UDP: 4789	The private IP address of mirror source **ECS-source**: 192.168.0.230/32	Allows UDP packets encapsulated by **ECS-source** to access **ECS-target-01** over port 4789.

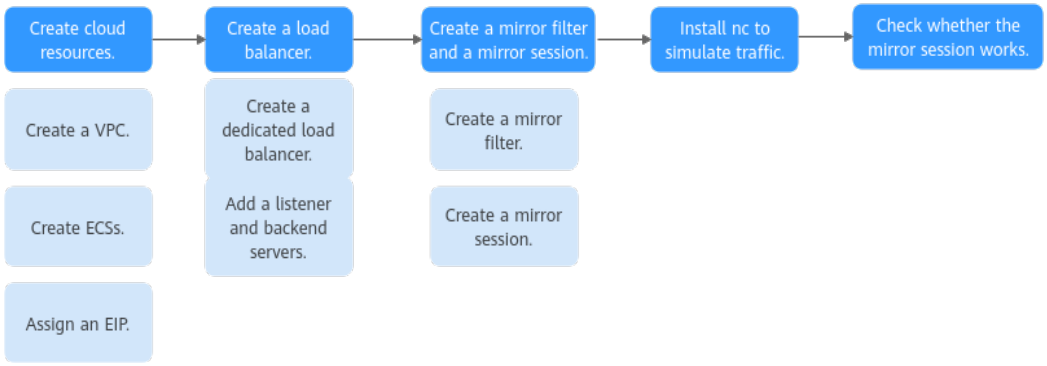
Table 13-43 Inbound and outbound rules of the mirror filter

Direction	Priority	Protocol	Action	Type	Source	Source Port Range	Destination	Destination Port Range
Inbound	1	TCP	Accept	IPv4	The private IP address of **ECS-test**: 192.168.0.161/32	All	The private IP address of **ECS-source**: 192.168.0.230/32	Port of **ECS-source**: 1234-1234
Outbound	1	TCP	Accept	IPv4	The private IP address of **ECS-source**: 192.168.0.230/32	Port of **ECS-source**: 1234-1234	The private IP address of **ECS-test**: 192.168.0.161/32	All

Procedure

Figure 13-13 shows the procedure required to mirror the inbound and outbound TCP traffic between a mirror source (network interface) and a given instance to a mirror target (load balancer).

Figure 13-13 Mirroring inbound and outbound TCP traffic to a load balancer



Step 1: Create Cloud Resources

1. Create a VPC and subnet.
   For details, see **Creating a VPC and Subnet**.
2. Create four ECSs.
   For details, see **Purchasing a Custom ECS**.
3. Assign an EIP.
   For details, see **Assigning an EIP**.

Step 2: Create a Load Balancer

1. Create a dedicated load balancer.
   For details, see **Creating a Dedicated Load Balancer**.
2. Add a UDP listener to the load balancer and add backend servers.
   For details, see **Adding a UDP Listener**.

Step 3: Create a Mirror Filter and a Mirror Session

1. Create a mirror filter.
   For details, see **Creating a Mirror Filter**.
2. Create a mirror session, and associate the mirror filter, mirror source, and mirror target with this mirror session.
   For details, see **Creating a Mirror Session**.

Step 4: Install Netcat (nc) to Simulate Traffic

The nc utility reads and writes data across network connections using TCP or UDP. It is usually used to test ports for accessibility. You need to install nc on both **ECS-source** and **ECS-test**.

1. Install nc on **ECS-source**.
   a. Bind the EIP to **ECS-source** to connect to the Internet for downloading the nc utility.
      For details, see **Binding an EIP to an ECS**.
   b. Remotely log in to **ECS-source**.
      For details, see **How Do I Log In to My ECS?**
   c. Run the following commands in sequence to install nc:
      sudo yum update
      Information similar to the following is displayed:

```
[root@ecs-source ~]# sudo yum update
HCE 2.0
base
        55 MB/s | 6.1 MB     00:00
HCE 2.0
updates
         98 MB/s |  14 MB     00:00
Last metadata expiration check: 0:00:01 ago on Tue 10 Sep 2024 05:54:28 PM CST.
Dependencies resolved.
Nothing to do.
Complete!
```

      sudo yum install nc

If information similar to the following is displayed, enter **y** as prompted and press **Enter**:

```
[root@ecs-source ~]# sudo yum install nc
Last metadata expiration check: 0:00:12 ago on Tue 10 Sep 2024 05:54:28 PM CST.
Dependencies resolved.
...
Install  2 Packages

Total download size: 6.1 M
Installed size: 25 M
Is this ok [y/N]: y
Downloading Packages:
...
Importing GPG key 0xA8DEF926:
 Userid     : "HCE <support@huaweicloud.com>"
 Fingerprint: C1BA 9CD4 9D03 A206 E241 F176 28DA 5B77 A8DE F926
 From       : http://repo.huaweicloud.com/hce/2.0/updates/RPM-GPG-KEY-HCE-2
Is this ok [y/N]: y
...
Installed:
  libssh2-1.10.0-2.r10.hce2.x86_64
nmap-2:7.92-2.r4.hce2.x86_64

Complete!
```

d. Unbind the EIP from **ECS-source** after nc is installed.

For details, see **Unbinding an EIP**.

2. Repeat **1.a** to **1.d** on **ECS-test**.
3. Release the EIP.

For details, see **Unbinding an EIP**. If you do not release the EIP, the EIP will continue to be billed.

Step 5: Check Whether the Mirror Session Works

1. Establish a TCP connection between **ECS-source** and **ECS-test**.

Use **ECS-source** to send TCP packets to **ECS-test** and check whether **ECS-test** can receive the packets.

a. Run the following command on **ECS-source** to listen to port 1234:

nc -l *<listening-port-of-mirror-source-ECS-source>*

Example command:

nc -l 1234

If the command output is empty, the port is opened for listening.

b. Run the following command on **ECS-test** to establish a TCP connection between **ECS-source** and **ECS-test**:

nc *<private-IP-address-of-mirror-source-ECS-source> <listening-port-of-mirror-source-ECS-source>*

Example command:

nc 192.168.0.230 1234

If the command output is empty, the TCP connection has been established.

c. Enter any information (for example, **hello**) on **ECS-source** and press **Enter** to check whether requests can be sent over the TCP connection.

```
[root@ecs-source ~]# nc -l 1234
hello
```


d. Check whether **ECS-test** can receive "hello" from **ECS-source**.

If information similar to the following is displayed, **ECS-test** can receive "hello" from **ECS-source**.

```
[root@ecs-test ~]# nc 192.168.0.230 1234
hello
```

2. Check whether the outbound TCP packet from **ECS-source** to **ECS-test** can be mirrored to backend servers **ECS-target-01** and **ECS-target-02** of **ELB-A**.

When **ECS-source** sends a TCP packet to **ECS-test**, run **tcpdump** to check whether **ECS-target-01** and **ECS-target-02** can receive the packet. If they do, the mirror session accepts the outbound TCP traffic.

a. Remotely log in to **ECS-target-01**.

For details, see **How Do I Log In to My ECS?**

b. Run the following command on **ECS-target-01** to view its network interface name:

ifconfig

Information similar to the following is displayed. In this example, the network interface of **ECS-target-01** is **eth0**.

```
[root@ecs-target-01 ~]# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.0.6  netmask 255.255.255.0  broadcast 192.168.0.255
        inet6 fe80::f816:3eff:fe7e:d6dc  prefixlen 64  scopeid 0x20<link>
        ether fa:16:3e:7e:d6:dc  txqueuelen 1000  (Ethernet)
        RX packets 87498  bytes 114570302 (109.2 MiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 18337  bytes 6613541 (6.3 MiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
...
```

c. Run the following command on **ECS-target-01** to check whether it can receive packets:

tcpdump -i *<network-interface-name-of-the-mirror-target>* **udp port 4789 -nne**

Example command:

tcpdump -i eth0 udp port 4789 -nne

Information similar to the following is displayed:

```
[root@ecs-target-01 ~]# tcpdump -i eth0 udp port 4789 -nne
dropped privs to tcpdump
tcpdump: verbose output suppressed, use -v[v]... for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), snapshot length 262144 bytes
```

d. Remotely log in to **ECS-target-02**.

For details, see **How Do I Log In to My ECS?**

e. Run the following command on **ECS-target-02** to view its network interface name:

ifconfig

Information similar to the following is displayed. In this example, the network interface of **ECS-target-02** is **eth0**.

```
[root@ecs-target-02 ~]# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.0.12  netmask 255.255.255.0  broadcast 192.168.0.255
        inet6 fe80::f816:3eff:fe7e:d6e2  prefixlen 64  scopeid 0x20<link>
        ether fa:16:3e:7e:d6:e2  txqueuelen 1000  (Ethernet)
        RX packets 87009  bytes 114283412 (108.9 MiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 17015  bytes 6492086 (6.1 MiB)
```



```
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
...
```

f. Run the following command on **ECS-target-02** to check whether it can receive packets:

tcpdump -i *<network-interface-name-of-the-mirror-target>* **udp port 4789 -nne**

Example command:

tcpdump -i eth0 udp port 4789 -nne

Information similar to the following is displayed:

```
[root@ecs-target-02 ~]# tcpdump -i eth0 udp port 4789 -nne
dropped privs to tcpdump
tcpdump: verbose output suppressed, use -v[v]... for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), snapshot length 262144 bytes
```

g. Enter any information (for example, **source to test**) on **ECS-source** and press **Enter** to send a TCP packet to **ECS-test**.

Information similar to the following is displayed:

```
[root@source ~]# nc -l 1234
hello
source to test
```

h. Check whether **ECS-test** can receive "source to test" from **ECS-source**.

If information similar to the following is displayed, **ECS-test** can receive "source to test" from **ECS-source**.

```
[root@ecs-test ~]# nc 192.168.0.230 1234
hello
source to test
```

i. Check whether **ECS-target-01** and **ECS-target-02** can receive the TCP packet.

- **ECS-target-01**

  If the information similar to the following is displayed, the packet containing "source to test" (time: **19:09:21.273376**) from **ECS-source** is not sent to **ECS-test** after running **tcpdump**. This means the accept rule works. **vni 1** is the identifier of **mirror-session-01**, indicating that **ECS-target-01** can receive the packet through this mirror session. The packet content has two parts: a VXLAN packet encapsulated by traffic mirroring and the original packet. For details, see **Table 13-27**.

  ```
  [root@ecs-target-01 ~]# tcpdump -i eth0 udp port 4789 -nne
  dropped privs to tcpdump
  tcpdump: verbose output suppressed, use -v[v]... for full protocol decode
  listening on eth0, link-type EN10MB (Ethernet), snapshot length 262144 bytes
  19:09:21.273376 fa:16:3e:d1:6b:5d > fa:16:3e:7e:d6:dc, ethertype IPv4 (0x0800), length 131:
  192.168.0.230.32816 > 192.168.0.6.4789: VXLAN, flags [I] (0x08), vni 1
  fa:16:3e:7e:d6:bc > fa:16:3e:7e:d6:77, ethertype IPv4 (0x0800), length 81:
  192.168.0.230.1234 > 192.168.0.161.57032: Flags [P.], seq 4181467553:4181467568, ack
  3509843935, win 510, options [nop,nop,TS val 476501697 ecr 998055381], length 15
  ```

- **ECS-target-02**

  Information similar to the following is displayed. You can view the response packet (time: **19:09:21.273429**) from **ECS-test** to **ECS-source** after running **tcpdump**.

  ```
  [root@ecs-target-02 ~]# tcpdump -i eth0 udp port 4789 -nne
  dropped privs to tcpdump
  tcpdump: verbose output suppressed, use -v[v]... for full protocol decode
  listening on eth0, link-type EN10MB (Ethernet), snapshot length 262144 bytes
  19:09:21.273429 fa:16:3e:d1:6b:5d > fa:16:3e:7e:d6:e2, ethertype IPv4 (0x0800), length
  116: 192.168.0.230.32805 > 192.168.0.12.4789: VXLAN, flags [I] (0x08), vni 1
  ```



```
fa:16:3e:7e:d6:77 > fa:16:3e:7e:d6:bc, ethertype IPv4 (0x0800), length 66:
192.168.0.161.57032 > 192.168.0.230.1234: Flags [.], ack 4181467568, win 502, options
[nop,nop,TS val 998154498 ecr 476501697], length 0
```

3. Check whether the inbound TCP packets from **ECS-test** to **ECS-source** can be mirrored to backend servers **ECS-target-01** and **ECS-target-02** of **ELB-A**.

   When **ECS-test** sends a TCP packet to **ECS-source**, use tcpdump to check whether **ECS-target-01** and **ECS-target-02** can receive the packet. If they do, the mirror session accepts the inbound TCP traffic.

   a. Enter any information (for example, **test to source**) on **ECS-test** and press **Enter** to send a TCP packet to **ECS-source**.

      Information similar to the following is displayed:

      ```
      [root@ecs-test ~]# nc 192.168.0.230 1234
      hello
      source to test
      test to source
      ```

   b. Check whether **ECS-source** can receive "test to source" from **ECS-test**.

      If information similar to the following is displayed, **ECS-source** can receive "test to source" from **ECS-test**.

      ```
      [root@ecs-source ~]# nc -l 1234
      hello
      source to test
      test to source
      ```

   c. Check whether **ECS-target-01** and **ECS-target-02** can receive the TCP packet.

      - **ECS-target-01**

        Information similar to the following is displayed. You can view the packet containing "test to source" (time: **19:10:04.772581**) from **ECS-test** to **ECS-source** after running **tcpdump**. This means the accept rule works. **vni 1** is the identifier of **mirror-session-01**, indicating that **ECS-target-01** can receive the packet through this mirror session. The packet content has two parts: a VXLAN packet encapsulated by traffic mirroring and the original packet. For details, see **Table 13-27**.

        ```
        [root@ecs-target-01 ~]# tcpdump -i eth0 udp port 4789 -nne
        dropped privs to tcpdump
        tcpdump: verbose output suppressed, use -v[v]... for full protocol decode
        listening on eth0, link-type EN10MB (Ethernet), snapshot length 262144 bytes
        19:09:21.273376 fa:16:3e:d1:6b:5d > fa:16:3e:7e:d6:dc, ethertype IPv4 (0x0800), length 131:
        192.168.0.230.32816 > 192.168.0.6.4789: VXLAN, flags [I] (0x08), vni 1
        fa:16:3e:7e:d6:bc > fa:16:3e:7e:d6:77, ethertype IPv4 (0x0800), length 81:
        192.168.0.230.1234 > 192.168.0.161.57032: Flags [P.], seq 4181467553:4181467568, ack
        3509843935, win 510, options [nop,nop,TS val 476501697 ecr 998055381], length 15
        19:10:04.772581 fa:16:3e:d1:6b:5d > fa:16:3e:7e:d6:dc, ethertype IPv4 (0x0800), length 131:
        192.168.0.230.32805 > 192.168.0.6.4789: VXLAN, flags [I] (0x08), vni 1
        fa:16:3e:7e:d6:77 > fa:16:3e:7e:d6:bc, ethertype IPv4 (0x0800), length 81:
        192.168.0.161.57032 > 192.168.0.230.1234: Flags [P.], seq 1:16, ack 15, win 502, options
        [nop,nop,TS val 998197997 ecr 476501697], length 15
        ```

      - **ECS-target-02**

        Information similar to the following is displayed. You can view the response packet (time: **19:10:04.772601**) from **ECS-test** to **ECS-source** after running **tcpdump**.

        ```
        [root@ecs-target-02 ~]# tcpdump -i eth0 udp port 4789 -nne
        dropped privs to tcpdump
        tcpdump: verbose output suppressed, use -v[v]... for full protocol decode
        listening on eth0, link-type EN10MB (Ethernet), snapshot length 262144 bytes
        19:09:21.273429 fa:16:3e:d1:6b:5d > fa:16:3e:7e:d6:e2, ethertype IPv4 (0x0800), length
        ```



```
116: 192.168.0.230.32805 > 192.168.0.12.4789: VXLAN, flags [I] (0x08), vni 1
fa:16:3e:7e:d6:77 > fa:16:3e:7e:d6:bc, ethertype IPv4 (0x0800), length 66:
192.168.0.161.57032 > 192.168.0.230.1234: Flags [.], ack 4181467568, win 502, options
[nop,nop,TS val 998154498 ecr 476501697], length 0
19:10:04.772601 fa:16:3e:d1:6b:5d > fa:16:3e:7e:d6:e2, ethertype IPv4 (0x0800), length
116: 192.168.0.230.32816 > 192.168.0.12.4789: VXLAN, flags [I] (0x08), vni 1
fa:16:3e:7e:d6:bc > fa:16:3e:7e:d6:77, ethertype IPv4 (0x0800), length 66:
192.168.0.230.1234 > 192.168.0.161.57032: Flags [.], ack 15, win 510, options [nop,nop,TS
val 476545196 ecr 998197997], length 0
```


14 Monitoring and Auditing

14.1 Cloud Eye Monitoring

14.1.1 Supported Metrics

Description

This section describes the namespace, list, and measurement dimensions of EIP and bandwidth metrics that you can check on Cloud Eye. You can use APIs or the Cloud Eye console to query the metrics of the monitored metrics and alarms generated for EIPs and bandwidths.

Namespace

SYS.VPC

Monitoring Metrics

Table 14-1 EIP and bandwidth metrics

ID	Name	Description	Value Range	Monitored Object	Monitoring Interval (Raw Data)
upstream_bandwidth	Outbound Bandwidth	Network rate of outbound traffic (Previously called "Upstream Bandwidth") Unit: bit/s	≥ 0 bit/s	Bandwidth or EIP	1 minute

ID	Name	Description	Value Range	Monitored Object	Monitoring Interval (Raw Data)
downstream_bandwidth	Inbound Bandwidth	Network rate of inbound traffic (Previously called "Downstream Bandwidth") Unit: bit/s	≥ 0 bit/s	Bandwidth or EIP	1 minute
upstream_bandwidth_usage	Outbound Bandwidth Usage	Usage of outbound bandwidth in the unit of percent. Outbound bandwidth usage = Outbound bandwidth/ Purchased bandwidth	0% to 100%	Bandwidth or EIP	1 minute
up_stream	Outbound Traffic	Network traffic going out of the cloud platform in a minute (Previously called "Upstream Traffic") Unit: byte	≥ 0 bytes	Bandwidth or EIP	1 minute
down_stream	Inbound Traffic	Network traffic going into the cloud platform in a minute (Previously called "Downstream Traffic") Unit: byte	≥ 0 bytes	Bandwidth or EIP	1 minute

Dimensions

Key	Value
publicip_id	EIP ID
bandwidth_id	Bandwidth ID

If a monitored object has multiple dimensions, all dimensions are mandatory when you use APIs to query the metrics.

- Query a monitoring metric:

  dim.0=bandwidth_id,530cd6b0-86d7-4818-837f-935f6a27414d&dim.1=publicip_id,3773b058-5b4f-4366-9035-9bbd9964714a
- Query monitoring metrics in batches:

  "dimensions": [

  {

  "name": "bandwidth_id",

  "value": "530cd6b0-86d7-4818-837f-935f6a27414d"

  }

  {

  "name": "publicip_id",

  "value": "3773b058-5b4f-4366-9035-9bbd9964714a"

  }

  ],

14.1.2 Viewing Metrics

Scenarios

You can view the bandwidth and EIP usage.

You can view the inbound bandwidth, outbound bandwidth, inbound bandwidth usage, outbound bandwidth usage, inbound traffic, and outbound traffic in a specified period.

Procedure (Cloud Eye Console)

1. Log in to the management console.
2. Click  in the upper left corner and select the desired region and project.
3. In the upper left corner of the page, click  to open the service list and choose **Management & Governance** > **Cloud Eye**.
4. Click **Cloud Service Monitoring** on the left of the page, and choose **Elastic IP and Bandwidth**.
5. Locate the target bandwidth or EIP and click **View Metric** in the **Operation** column to check the bandwidth or EIP monitoring information.

14.1.3 Creating an Alarm Rule

Scenarios

You can configure alarm rules to customize the monitored objects and notification policies. You can learn your resource statuses at any time.

Procedure

1. Log in to the management console.
2. Click  in the upper left corner and select the desired region and project.
3. In the upper left corner of the page, click  to open the service list and choose **Management & Governance** > **Cloud Eye**.
4. In the left navigation pane on the left, choose **Alarm Management** > **Alarm Rules**.
5. On the **Alarm Rules** page, click **Create Alarm Rule** and set required parameters, or modify an existing alarm rule.
6. After the parameters are set, click **Create**.

   After the alarm rule is created, the system automatically notifies you if an alarm is triggered for the VPC service.

   NOTE

   For more information about alarm rules, see **Cloud Eye User Guide**.

14.2 CTS Auditing

14.2.1 Key Operations Recorded by CTS

With CTS, you can record operations performed on the VPC service for further query, audit, and backtracking purposes.

Table 14-2 lists the VPC operations that can be recorded by CTS.

Table 14-2 VPC operations that can be recorded by CTS

Operation	Resource Type	Trace Name
Modifying a bandwidth	Bandwidth	modifyBandwidth
Assigning an EIP	EIP	createEip
Releasing an EIP	EIP	deleteEip
Binding an EIP	EIP	bindEip
Unbinding an EIP	EIP	unbindEip
Assigning a private IP address	Private IP address	createPrivateIp

Operation	Resource Type	Trace Name
Deleting a private IP address	Private IP address	deletePrivateIp
Creating a security group	security_groups	createSecurity-group
Updating a security group	security_groups	updateSecurity-group
Deleting a security group	security_groups	deleteSecurity-group
Creating a security group rule	security-group-rules	createSecurity-group-rule
Updating a security group rule	security-group-rules	updateSecurity-group-rule
Deleting a security group rule	security-group-rules	deleteSecurity-group-rule
Creating a subnet	Subnet	createSubnet
Deleting a subnet	Subnet	deleteSubnet
Modifying a subnet	Subnet	modifySubnet
Creating a VPC	VPC	createVpc
Deleting a VPC	VPC	deleteVpc
Modifying a VPC	VPC	modifyVpc
Creating a VPN	VPN	createVpn
Deleting a VPN	VPN	deleteVpn
Modifying a VPN	VPN	modifyVpn
Creating a router	routers	createRouter
Updating a router	routers	updateRouter
Adding an interface to a router	routers	addRouterInterface
Deleting an interface from a router	routers	removeRouterInterface
Creating a port	ports	createPort
Updating a port	ports	updatePort
Deleting a port	ports	deletePort
Creating a network	networks	createNetwork
Updating a network	networks	updateNetwork

Operation	Resource Type	Trace Name
Deleting a network	networks	deleteNetwork
Batch creating or deleting subnet tags	tag	batchUpdateTags
Batch creating or deleting VPC tags	tag	batchUpdateVpcTags
Creating a route table	routetables	createRouteTable
Updating a route table	routetables	updateRouteTable
Deleting a route table	routetables	deleteRouteTable
Creating a VPC peering connection	vpc-peerings	createVpcPeerings
Updating a VPC peering connection	vpc-peerings	updateVpcPeerings
Deleting a VPC peering connection	vpc-peerings	deleteVpcPeerings
Creating a network ACL group	firewall-groups	createFirewallGroup
Updating a network ACL group	firewall-groups	updateFirewallGroup
Deleting a network ACL group	firewall-groups	deleteFirewallGroup
Creating a network ACL policy	firewall-policies	createFirewallPolicy
Updating a network ACL policy	firewall-policies	updateFirewallPolicy
Deleting a network ACL policy	firewall-policies	deleteFirewallPolicy
Inserting a network ACL rule	firewall-policies	insertFirewallPolicyRule
Removing a network ACL rule	firewall-policies	removeFirewallPolicyRule
Creating a network ACL rule	firewall-rules	createFirewallRule
Updating a network ACL rule	firewall-rules	updateFirewallRule
Deleting a network ACL rule	firewall-rules	deleteFirewallRule

Operation	Resource Type	Trace Name
Creating an IP address group	address_group	createAddress_group
Updating an IP address group	address_group	updateAddress_group
Forcibly deleting an IP address group	address_group	force_deleteAddress_group
Deleting an IP address group	address_group	deleteAddress_group
Creating a flow log	flowlogs	createFlowLog
Updating a flow log	flowlogs	updateFlowLog
Deleting a flow log	flowlogs	deleteFlowLog
Creating a public NAT gateway	natgateway	createNatGateway
Modifying a public NAT gateway	natgateway	updateNatGateway
Deleting a public NAT gateway	natgateway	deleteNatGateway
Creating a DNAT rule	dnatrule	createDnatRule
Modifying a DNAT rule	dnatrule	updateDnatRule
Deleting a DNAT rule	dnatrule	deleteDnatRule
Creating an SNAT rule	snatrule	createSnatRule
Modifying an SNAT rule	snatrule	updateSnatRule
Deleting an SNAT rule	snatrule	deleteSnatRule

14.2.2 Viewing Traces

Scenarios

After CTS is enabled, CTS starts recording operations on cloud resources. The CTS management console stores the last seven days of operation records.

This section describes how to query or export the last seven days of operation records on the management console.

NOTICE

CTS only retains traces for seven days. To store traces for a longer time, configure your tracker to transfer traces to OBS buckets. For details, see **Configuring a Tracker**.

Procedure

1. Log in to the management console.
2. Click  in the upper left corner and select the desired region and project.
3. In the upper left corner of the page, click  to go to the service list. Under **Management & Governance**, click **Cloud Trace Service**.
4. In the navigation pane on the left, choose **Trace List**.
5. Specify filters as needed. The following filters are available:
   - **Trace Type**: Set it to **Management** or **Data**.
   - **Trace Source**, **Resource Type**, and **Search By**

     Select filters from the drop-down list.

     If you select **Trace name** for **Search By**, select a trace name.

     If you select **Resource ID** for **Search By**, select or enter a resource ID.

     If you select **Resource name** for **Search By**, select or enter a resource name.
   - **Operator**: Select a specific operator (a user other than an account).
   - **Trace Status**: Select **All trace statuses**, **Normal**, **Warning**, or **Incident**.
   - Search time range: In the upper right corner, choose **Last 1 hour**, **Last 1 day**, or **Last 1 week**, or specify a custom time range.
6. Click arrow on the left of the required trace to expand its details.
7. Locate the required trace and click **View Trace** in the **Operation** column.

   A dialog box is displayed, showing the trace content.

15 Managing Quotas

What Is a Quota?

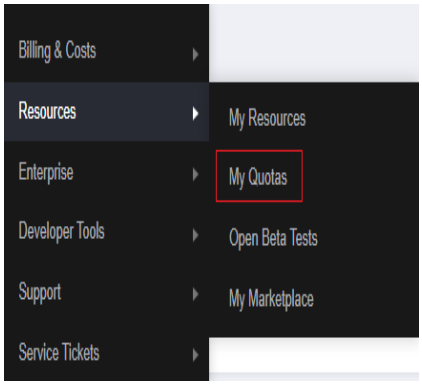
A quota limits the quantity of a resource available to users, thereby preventing spikes in the usage of the resource. For example, a VPC quota limits the number of VPCs that can be created.

You can also request for an increased quota if your existing quota cannot meet your service requirements.

How Do I View My Quotas?

1. Log in to the management console.
2. Click in the upper left corner and select the desired region and project.
3. In the upper right corner of the page, choose **Resources** > **My Quotas**.
   The **Service Quota** page is displayed.

   Figure 15-1 My Quotas



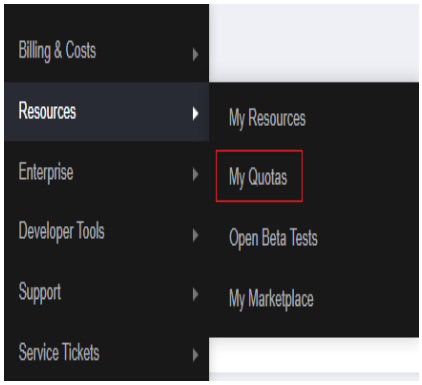
4. View the used and total quota of each type of resources on the displayed page.
   If a quota cannot meet service requirements, apply for a higher quota.

How Do I Apply for a Higher Quota?

1. Log in to the management console.

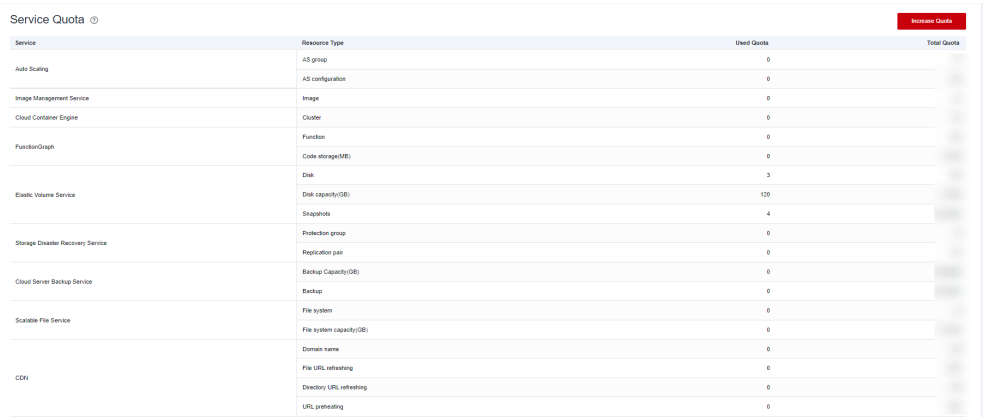
2. In the upper right corner of the page, choose **Resources** > **My Quotas**.
   The **Service Quota** page is displayed.

   Figure 15-2 My Quotas



3. Click **Increase Quota** in the upper right corner of the page.

   Figure 15-3 Increasing quota



4. On the **Create Service Ticket** page, configure parameters as required.
   In the **Problem Description** area, fill in the content and reason for adjustment.
5. After all necessary parameters are configured, select **I have read and agree to the Ticket Service Protocol and Privacy Statement** and click **Submit**.

