虚拟私有云

最佳实践

文档版本	01
发布日期	2024-07-22

华为云计算技术有限公司

商标声明


和其他华为商标均为华为技术有限公司的商标。
本文档提及的其他所有商标或注册商标，由各自的所有人拥有。

注意

您购买的产品、服务或特性等应受华为云计算技术有限公司商业合同和条款的约束，本文档中描述的全部或部分产品、服务或特性可能不在您的购买或使用范围之内。除非合同另有约定，华为云计算技术有限公司对本文档内容不做任何明示或暗示的声明或保证。

由于产品版本升级或其他原因，本文档内容会不定期进行更新。除非另有约定，本文档仅作为使用指导，本文档中的所有陈述、信息和建议不构成任何明示或暗示的担保。

华为云计算技术有限公司

地址： 贵州省贵安新区黔中大道交兴功路华为云数据中心 邮编：550029

网址： https://www.huaweicloud.com/

目 录

1 Solution as Code 一键式部署类最佳实践

为帮助企业高效上云，华为云Solution as Code萃取丰富上云成功实践，提供一系列基于华为云可快速部署的解决方案，帮助用户降低上云门槛。同时开放完整源码，支持个性化配置，解决方案开箱即用，所见即所得。

表 1-1 Solution as Code 一键式部署类最佳实践汇总

一键式部署方案	说明	相关服务
基于VPCEP实现跨VPC连接ELB	该解决方案基于VPC终端节点VPCEP和终端节点服务，帮助用户快速实现同一区域不经过公网、跨虚拟私有云 VPC的弹性负载均衡 ELB后端服务访问	VPC、ECS、ELB、VPCEP
基于SNAT实现公网访问解决方案	该解决方案能帮用户快速实现多个无弹性公网IP的云主机安全访问互联网，轻松构建VPC的公网出口	VPC、ECS、NAT、EIP

2 虚拟私有云和子网规划建议

当您需要使用虚拟私有云VPC和子网搭建您的云上网络时，请您参考以下规划建议，并结合实际业务情况，规划您的VPC和子网数量、VPC和子网IP网段。同时，如果需要连通不同VPC网络，或者连通VPC和云下数据中心网络时，需要额外注意通信的网段之间不能冲突。合理规划VPC和子网，可以避免网段临时扩容或者IP网段冲突可能导致的问题。详细规划建议，请您参见以下内容：

- **如何规划VPC的数量?**
- **如何规划子网的数量?**
- **如何规划VPC和子网的IP网段?**
- **如何规划路由表的数量?**
- **当VPC与其他VPC通信或者VPC与云下数据中心通信时如何规划网络?**

如何规划 VPC 的数量?

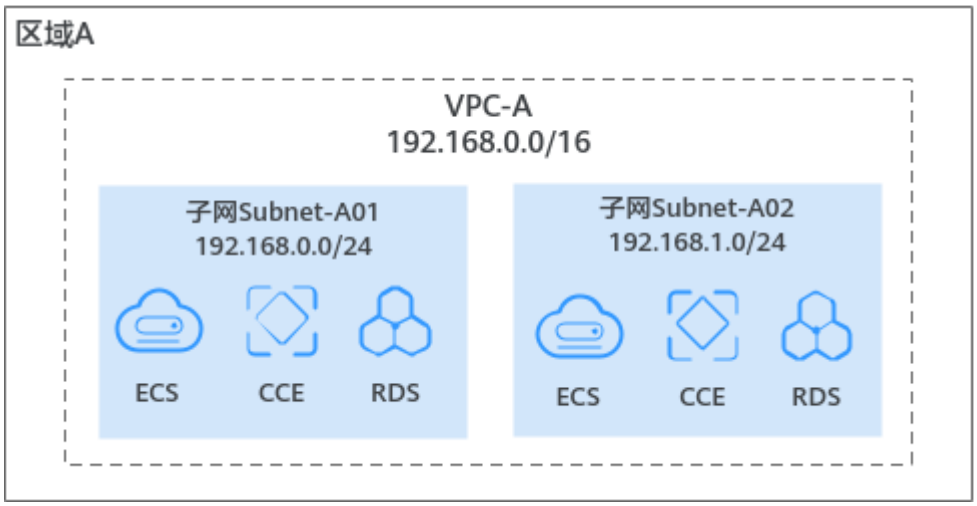
VPC具有区域属性，VPC内的云资源（比如ECS、CCE、RDS等）必须和VPC位于同一个区域内。默认情况下，不同VPC之间网络隔离，同一个VPC内的不同子网之间网络互通。

规划一个 VPC

如果您的业务部署在一个区域，并且业务量不大，不同业务之间不需要网络隔离，那么推荐你规划一个VPC。

您可以在一个VPC中创建多个子网和路由表。子网可以将VPC网段划分成若干段，不同子网承载不同的业务。同时，您还可以将不同子网关联至不同的路由表，灵活控制子网的网络流量。如**图2-1**所示，在区域A内，业务部署在VPC-A内的不同子网。

图 2-1 规划 1 个 VPC



规划多个 VPC

当您的业务有以下任意一个需求时，则一个VPC无法满足业务要求，推荐您规划多个VPC。

- **业务需要部署在多个区域**

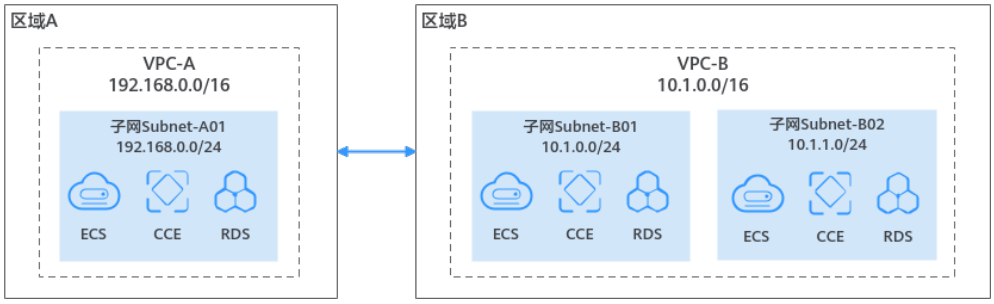
  VPC是区域级别的服务，一个VPC无法实现跨区域部署业务。如果您的业务同时部署在多个区域，则在每个区域下，至少需要规划一个VPC。

  不同VPC之间网络隔离，您可以搭配网络连通服务连通不同VPC的网络。

  - 连通相同区域内的不同VPC：您可以使用**对等连接**或者**企业路由器**来实现。
  - 连通不同区域内的VPC：您可以使用**云连接**来实现。

  如**图2-2**所示，一部分业务部署在区域A内的VPC-A中，一部分业务部署在区域B内的VPC-B中，通过云连接连通VPC-A和VPC-B的网络。

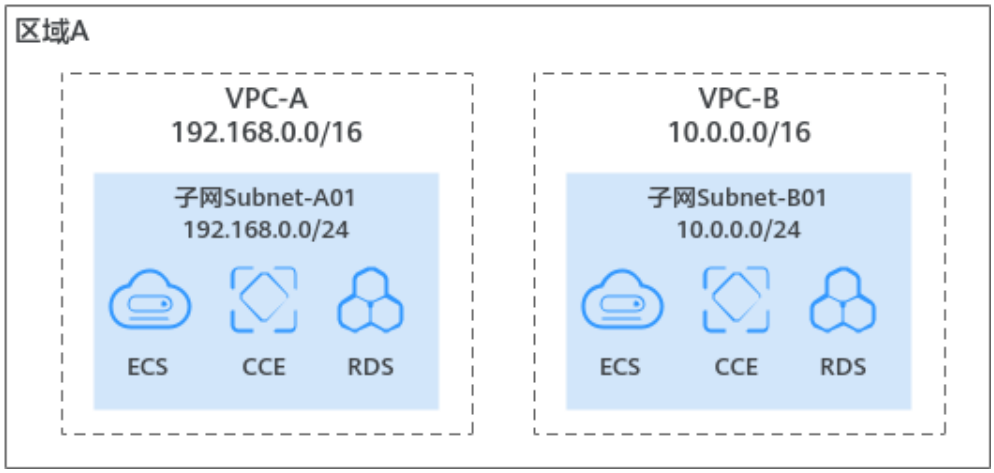
  图 2-2 规划多个 VPC（业务需要部署在多个区域）

- **业务部署在一个区域且网络隔离**

  如果您的业务部署在一个区域，并且不同业务之间网络隔离，则您需要在同一个区域下规划多个VPC，由于不同VPC之间网络隔离，则每个业务独立部署在一个VPC上即可满足要求。如**图2-3**所示，在区域A内，一部分业务部署在VPC-A中，一部分业务部署在VPC-B中，两个VPC之间网络隔离。

图 2-3 规划多个 VPC（业务部署在一个区域且网络隔离）



说明

一个用户在单个区域可创建的虚拟私有云数量默认为5个，如果您需要提升配额，请参见**如何申请扩大配额?**

如何规划子网的数量?

子网是VPC内的IP地址集，可以将VPC的网段分成若干块，子网划分可以帮助您合理规划IP地址资源。VPC中的所有云资源都必须部署在子网内。

通常情况下，部署在同一个VPC内的业务，您可以根据业务模块来划分子网，比如在VPC-A内，子网A01用于Web层，子网A02用于管理层，子网A03用于数据层。根据业务划分子网模块，有利于结合网络ACL进行网络防护。

关于子网和云资源可用区的选择，您还需要了解以下原则：

- 默认情况下，同一个VPC中，不同子网内的所有实例网络互通。同一个VPC内的子网可以位于不同可用区，不影响通信。比如VPC-A内有子网A01（可用区A）和子网A02（可用区B），子网A01和子网A02的网络默认互通。
- 同时，使用子网的云资源，其可用区和子网的可用区不用保持一致。比如位于可用区1的云服务器，可以使用可用区3的子网。假如可用区3发生故障，此时可用区1的云服务器可以继续使用可用区3的子网，不会影响云服务器上部署的业务。

说明

一个用户在单个区域可创建的子网数量默认为100个，如果您需要提升配额，请参见**如何申请扩大配额?**

如何规划 VPC 和子网的 IP 网段?

VPC和子网创建完成后，则无法修改网段。因此创建VPC和子网之前，请您务必结合业务规模和通信需求，合理规划VPC和子网网段，以便于业务的平滑扩展和运维。

说明

私有网络支持IPv4和IPv6网段地址。您可以自定义IPv4网段，不支持自定义IPv6网段，系统自动为每个子网分配一个掩码为64位IPv6网段，比如2407:c080:802:1b32::/64。

规划 VPC 网段

创建VPC的时候，您需要为VPC指定IPv4网段。VPC网段的选择需要考虑以下原则：

- IP地址数量：要为业务预留足够的IP地址，防止业务扩展给网络带来冲击。
- IP地址网段：当您要创建多个VPC，并且VPC与其他VPC、或者VPC与云下数据中心需要通信时，要避免网络两端的网段冲突，否则无法正常通信。

创建VPC时，您配置的IPv4网段是VPC的主网段。当VPC创建完成后，主网段不支持修改，若主网段不够分配，您可以**为VPC添加IPv4扩展网段**。

在创建VPC的时候，建议您使用**RFC 1918**中指定的私有IPv4地址范围，作为VPC的网段，具体如**表2-1**所示。

表 2-1 VPC 网段（RFC 1918）

VPC网段	IP地址范围	掩码范围	VPC网段示例
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

除了上述地址，您还可以使用任何可公共路由的IPv4地址（非RFC 1918指定的私有IPv4地址范围），但是必须排除**表2-2**中的系统预留地址和公网保留地址：

表 2-2 系统预留地址和公网保留地址

系统预留地址	公网保留地址
• 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

规划子网网段

- 子网掩码规划：子网的网段必须在VPC网段范围内，同一个VPC内的子网网段不可重复。子网网段的掩码长度范围是：所在VPC掩码~29，比如VPC网段为10.0.0.0/16，VPC的掩码为16，则子网的掩码可在16~29范围内选择。
- 子网内可用IP数量：子网创建成功后，不支持修改网段，请您结合业务所需的IP地址数量，提前合理规划好子网网段。
  - 子网网段不能太小，需要确保子网内可用IP地址数量可以满足业务需求。子网网段中第一个地址和后三个地址为系统预留地址，不能供实际业务使用，比如子网（10.0.0.0/24）中，10.0.0.1为网关地址、10.0.0.253为系统接口、10.0.0.254为DHCP使用、10.0.0.255为广播地址。

  - 子网网段也不能太大，以免后续扩展新的业务时，VPC内可用网段不够再创建新的子网。
- 子网网段避免冲突：如果子网所在的VPC与其他VPC、或者VPC与云下数据中心需要通信时，则VPC子网网段和网络对端网段不能相同，否则无法正常通信。

  如果网络两端的子网网段已经相同，您可以创建新的子网，请参见**为虚拟私有云创建新的子网**。

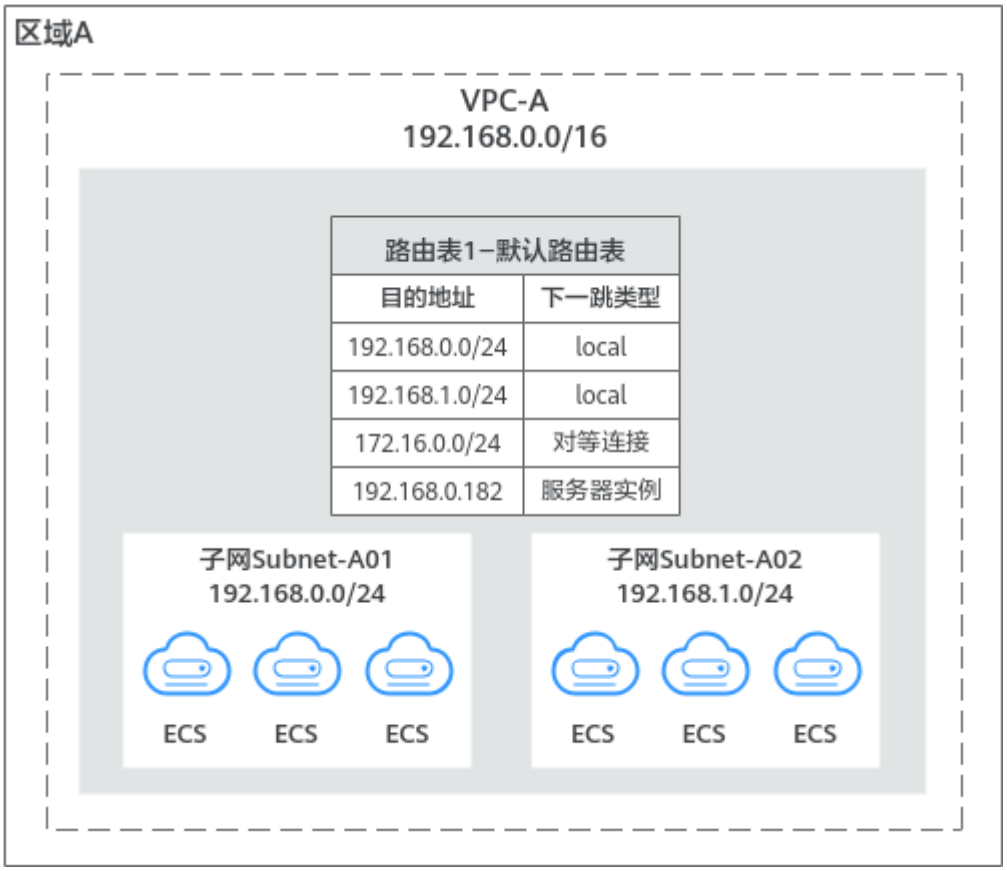
如何规划路由表的数量?

路由表由一系列路由规则组成，路由规则包括流量的目的地址和下一跳等信息，用于控制VPC内子网的出流量走向。一个VPC内可以拥有多个路由表，请您参考以下建议规划路由表。

规划一个路由表

当VPC内不同子网的网络流量走向需求相同或者差异不大，则推荐您规划一个路由表。用户创建VPC时，系统会自动生成一个默认路由表，子网会自动关联默认路由表，您可以在默认路由表中添加不同的路由来控制流量走向。如**图2-4**所示，VPC-A中只有一个默认路由表，子网Subnet-A01和Subnet-A02均关联至默认路由表。

图 2-4 规划一个路由表

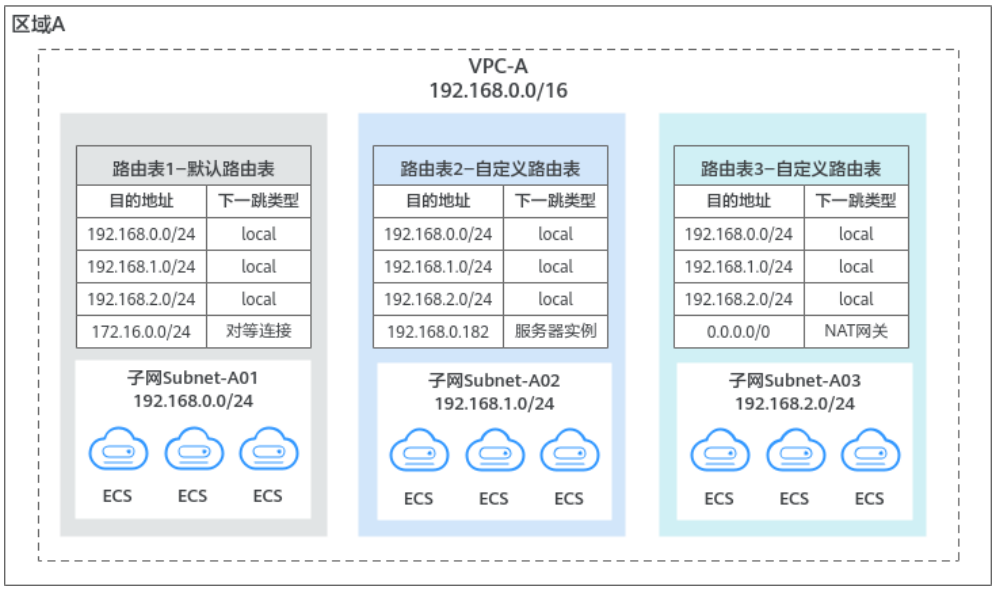


规划多个路由表

当VPC内不同子网的网络流量走向差异较大，则一个默认路由表无法满足业务需求，此时推荐您创建自定义路由表，将不同的子网关联至不同的路由表，实现不同流量走

向的控制。如图2-5所示，VPC-A中有三个路由表，子网Subnet-A01关联至路由表1（默认路由表），子网Subnet-A02关联至路由表2（自定义路由表），子网Subnet-A03关联至路由表3（自定义路由表）。

图 2-5 规划多个路由表



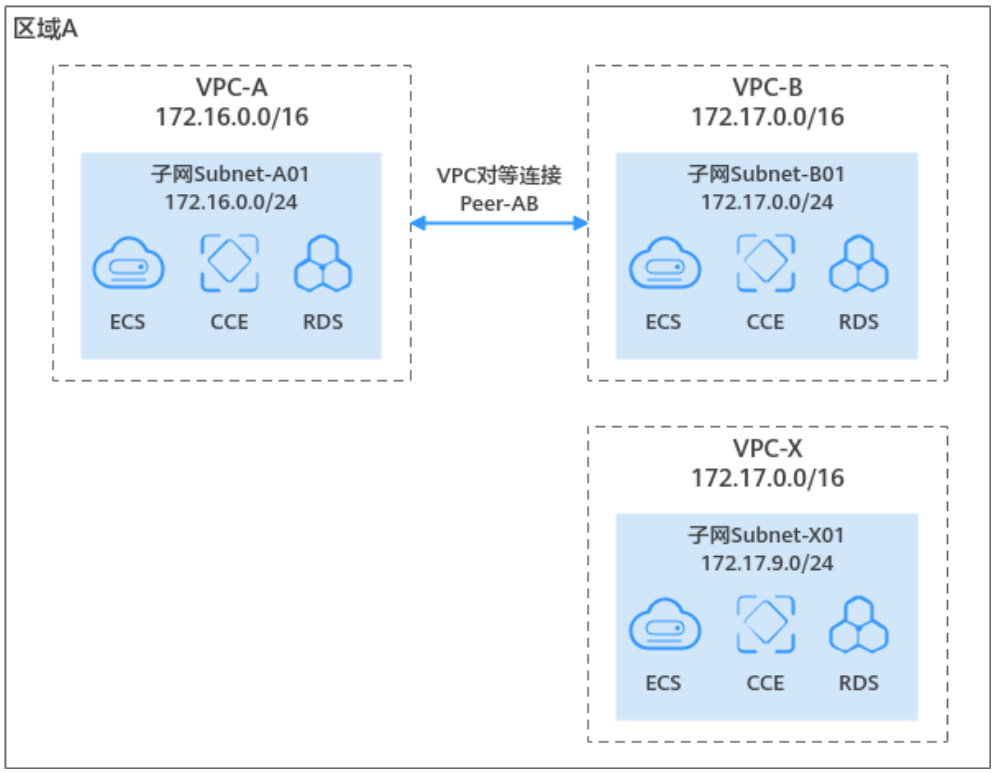
当 VPC 与其他 VPC 通信或者 VPC 与云下数据中心通信时如何规划网络?

如果您有VPC与其他VPC通信，或者VPC与云下数据中心通信的需求时，请确保VPC网段和需要通信的对端网段没有冲突。以下为您提供典型组网的网段规划示例。

连通 VPC 与其他 VPC

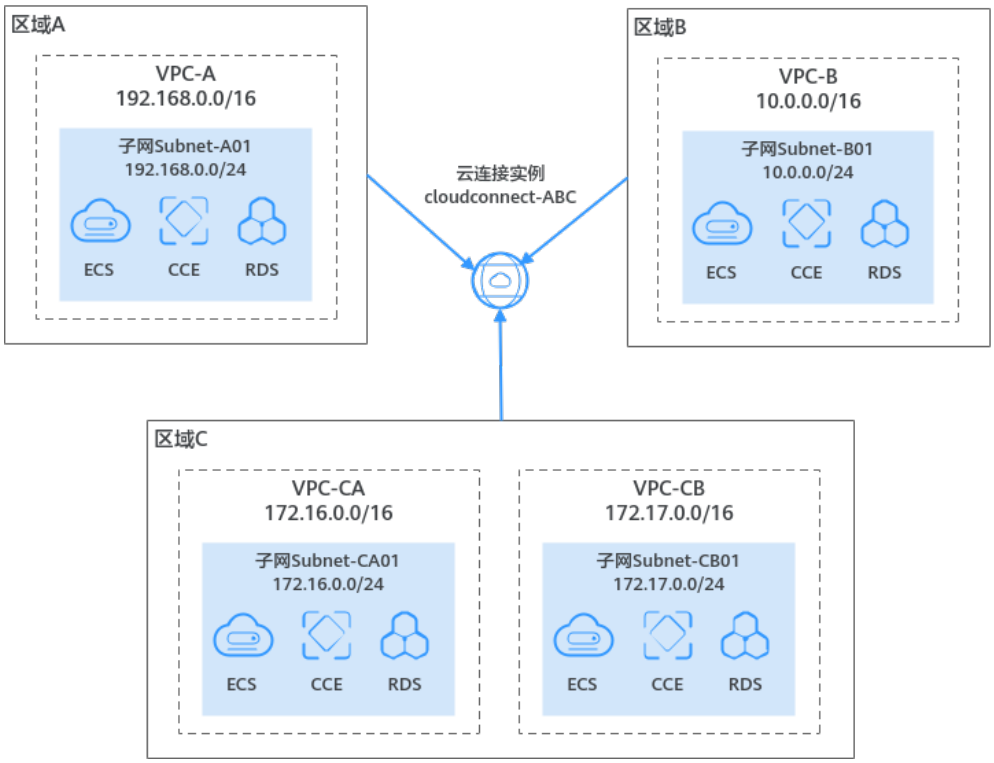
- 连通同区域的VPC：如图2-6所示，在区域A内，一共有三个VPC，分别为VPC-A、VPC-B和VPC-X。由于业务需求，需要连通VPC-A和VPC-B的网络，VPC-X不需要和其他VPC连通。
  - 由于VPC-A和VPC-B需要通信，则VPC-A和VPC-B的网段不能相同，通过对等连接连通VPC之间的内网网络。
  - 由于VPC-X和其他VPC之间不需要连通，因此VPC-X的网段可以和VPC-B相同，当前不会影响通信。但是基于业务的变化考虑，如果后续VPC-X和VPC-B需要通信，则在网段相同的基础上，建议VPC-B和VPC-X内的子网网段不能相同，则可以建立子网之间的对等连接。

图 2-6 连通同区域的 VPC



- 连通不同区域的VPC：如图2-7所示，业务需要部署在三个不同的区域内的VPC，分别为VPC-A、VPC-B、VPC-CA和VPC-CB。使用云连接可以快速连通不同区域的VPC，VPC基于云内骨干网络实现内网通信，需要通信的VPC网段不能相同。

图 2-7 连通不同区域的 VPC

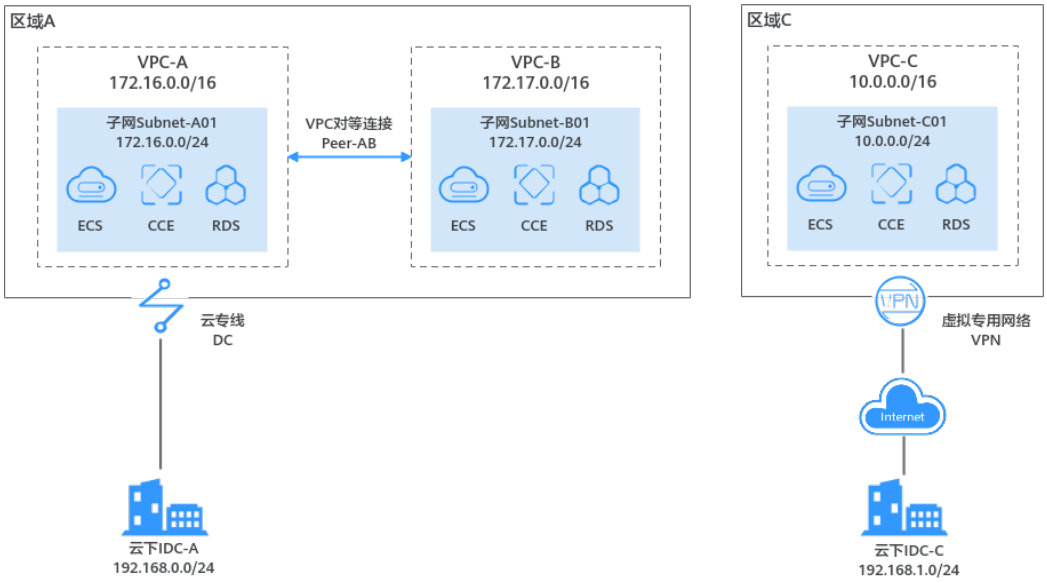


连通 VPC 和云下数据中心

如图2-8所示，在区域A内，VPC-A和VPC-B之间需要互通，并且VPC-A需要连通云下数据中心IDC-A。在区域C内，VPC-C需要连通云下数据中心IDC-C。

- 在区域A内，VPC-A和VPC-B的网段不同，可以通过对等连接连通网络。VPC-A和IDC-A通过云专线连通，VPC-A和IDC-A的网段不能相同。
- 在区域C内，VPC-C和IDC-C使用VPN通过互联网连通，VPC-C和IDC-C的网段不能相同。

图 2-8 连通 VPC 和云下数据中心



相关文档

- 您可以通过VPC快速搭建一个具有IPv4地址段的云上私有网络，同时，还可以通过EIP实现云上网络和公网通信的需求，具体请参见**通过VPC快速搭建IPv4网络**。
- 您可以通过VPC快速搭建一个同时具有IPv4和IPv6地址段的云上私有网络。同时，还可以通过EIP和共享带宽，实现IPv4和IPv6公网通信需求，具体请参见**通过VPC快速搭建IPv4/IPv6双栈网络**。

3 VPC 连接

访问 Internet

VPC内的云资源连接公网（Internet），可以通过如下云产品实现。

表 3-1 连接公网

云产品	应用场景	描述	相关操作
弹性公网IP	单个ECS连接公网	申请一个弹性公网IP（EIP）并将其绑定到ECS上，ECS即可连接公网，实现主动访问公网或面向公网提供服务。 支持动态绑定和解绑ECS。 可以使用共享带宽和共享流量包，降低公网成本。	**使用EIP连接公网**
NAT网关	多个ECS共享弹性公网IP连接公网	NAT网关提供SNAT和DNAT两种功能：SNAT可实现同一VPC内的多个ECS共享一个或多个EIP主动访问公网，有效降低管理成本，同时减少了ECS的EIP直接暴露的风险。DNAT功能还可以实现端口级别的转发，将EIP的端口映射到不同ECS的端口上，使VPC内多个ECS共享同一EIP和带宽面向公网提供服务，但没有均衡流量的功能。	**使用SNAT连接公网** **使用DNAT面向公网提供服务**
弹性负载均衡	通过将访问流量均衡分发到多个ECS的方式对外提供服务，比如电商等高并发访问场景	弹性负载均衡（ELB）可以将访问流量均衡分发（支持4层和7层两种方式）到多个后端ECS上，通过绑定EIP支撑海量用户从公网访问ECS。 通过流量分发扩展应用系统对外的服务能力，通过消除单点故障提升应用系统的可用性。	**弹性负载均衡介绍**

云上 VPC 互连

VPC与VPC之间要建立连接，可以通过如下云产品实现。

表 3-2 连接 VPC

云产品	应用场景	描述	相关操作
对等连接	同区域的VPC互连	对于同一区域的VPC，可以通过对等连接进行互连，同一账号与不同账号的连接方式略有差异。对等连接免费。	**创建同一账户下的对等连接** **创建不同账户下的对等连接**
云连接	跨区域的VPC互连	对于不同区域的VPC，不区分是否同一账号，都可以互连，跨区域连接实现全球云上网络。	**跨区域VPC互通**
虚拟专用网络VPN	使用公网低成本连接跨区域VPC	基于Internet使用加密隧道将不同区域的VPC连接起来。具备成本低、配置简单、即开即用等优点。但它的网络质量依赖Internet。	**通过VPN连接VPC**

连接线下数据中心（IDC）

对于自建本地数据中心（IDC）的用户，由于利旧和平滑演进的原因，并非所有的业务都能放置在云上，这个时候就可以通过如下产品构建混合云，实现云上VPC与云下IDC之间的互连。

表 3-3 连接 IDC

云产品	应用场景	描述	相关操作
虚拟专用网络VPN	使用公网低成本连接VPC与本地IDC	基于Internet使用加密隧道将VPC与本地数据中心连接起来。具备成本低、配置简单、即开即用等优点。但它的网络质量依赖Internet。	**通过VPN连接VPC**
云专线	铺设物理专线高质量连接VPC与本地IDC	使用物理专线将VPC与本地数据中心连接起来。具备低时延、高安全、专用等优点。适用对网络传输质量和安全等级要求较高的场景 。	**通过用户专线访问多个VPC**
云连接	跨区域的VPC、IDC互连	将要互通的本地IDC关联的云专线加载到已创建的云连接实例中，实现跨区域的VPC、IDC互连。	**多数据中心与多区域VPC互通**

4 私网访问

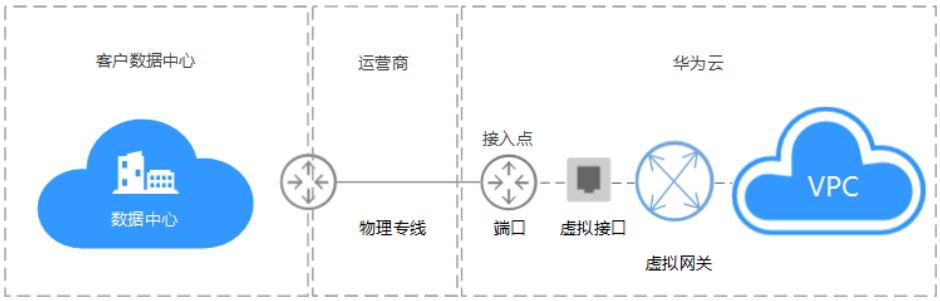
连接本地 IDC

您可以将云上VPC和本地IDC网络连接起来构建混合云。通过VPC和本地IDC之间安全可靠的连接，并借助华为云海量的计算、存储、网络资源，您可按需按量实时地将本地的IT基础架构无缝地扩展到华为云上，以应对业务波动。云专线、VPN都可以实现本地IDC和云上VPC互连。

- 云专线

  云专线（Direct Connect）用于搭建企业自有计算环境到华为云用户VPC环境的高速、稳定、安全的专属通道。您可使用云专线将本地数据中心的计算机与华为云上的云服务器或托管主机实现私网互连，充分利用云计算优势的同时，继续使用现有设施，实现灵活一体，可伸缩的混合IT计算环境。

  图 4-1 使用云专线连接本地 IDC

- VPN

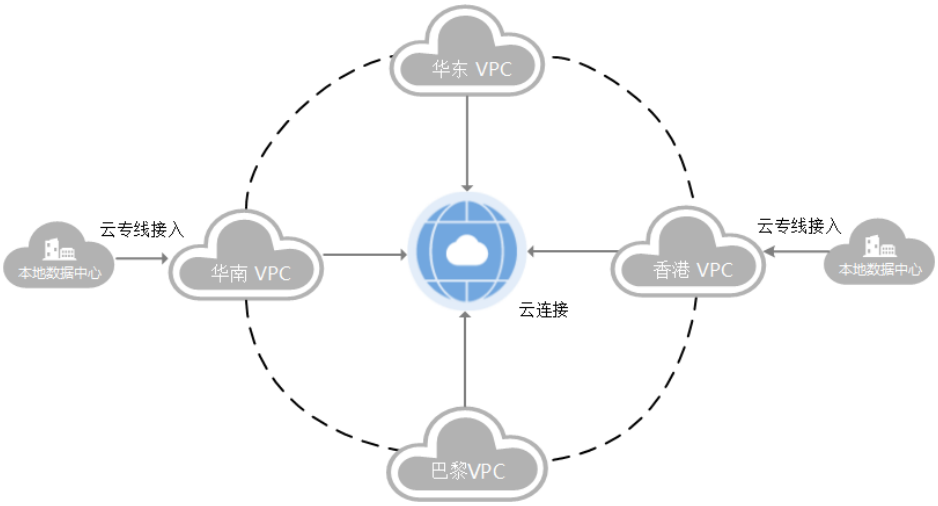
  虚拟专用网络（Virtual Private Network，简称VPN），用于在远端用户和VPC之间建立一条安全加密的公网通信隧道。当您作为远端用户需要访问VPC的业务资源时，您可以通过VPN连通VPC。

多站点互连

您可以通过云连接（Cloud Connect）实现全球多区域、多数据中心间的互连。

云连接为用户提供一种能够快速构建跨区域VPC之间以及云上多VPC与云下多数据中心之间的高速、优质、稳定的网络能力，帮助用户打造一张具有企业级规模和通信能力的全球云上网络。

图 4-2 使用云连接实现多站点互连



VPC 互连

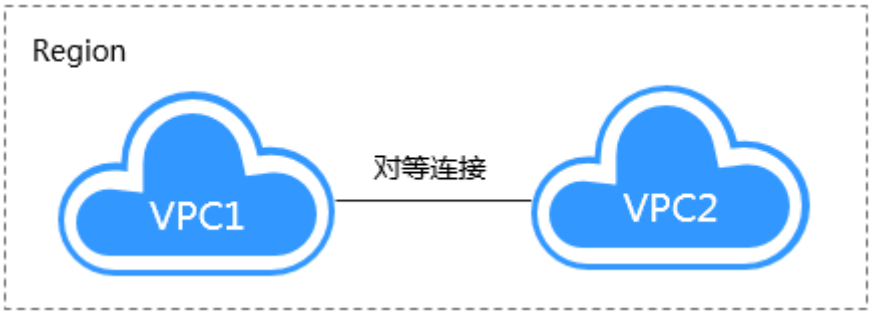
当您需要将相同区域不同的VPC之间互连时，推荐通过VPC对等连接来实现。

当您需要将不同区域的VPC之间互连并构建多区域服务网络时，推荐通过云专线、VPN、云连接服务来实现。

- 对等连接

  对于同一区域的不同VPC，可以通过对等连接进行互连。

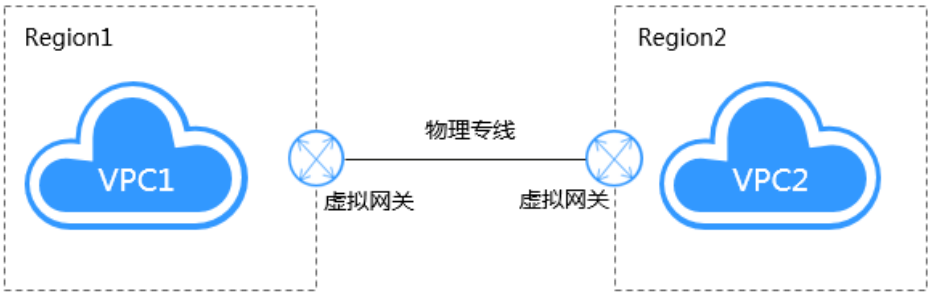
  图 4-3 使用对等连接实现同一区域 VPC 互连

- 云专线

  云专线（Direct Connect）用于搭建企业自有计算环境到华为云用户VPC环境的高速、稳定、安全的专属通道。您可使用云专线将本地数据中心的计算机与华为云上的云服务器或托管主机实现私网相连，充分利用云计算优势的同时，继续使用现有设施，实现灵活一体，可伸缩的混合IT计算环境。云专线也可以用于不同区域的VPC互连。

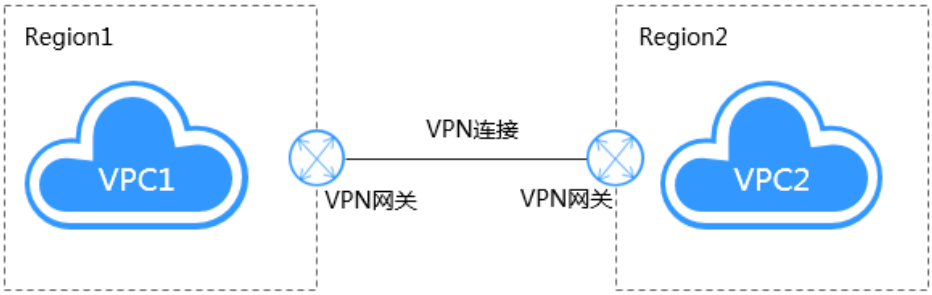
图 4-4 使用云专线实现不同区域 VPC 互连



- VPN

  VPN用于在远端用户和VPC之间建立一条安全加密的公网通信隧道。当您作为远端用户需要访问VPC的业务资源时，您可以通过VPN连通VPC。VPN也可以用于不同区域的VPC互连。

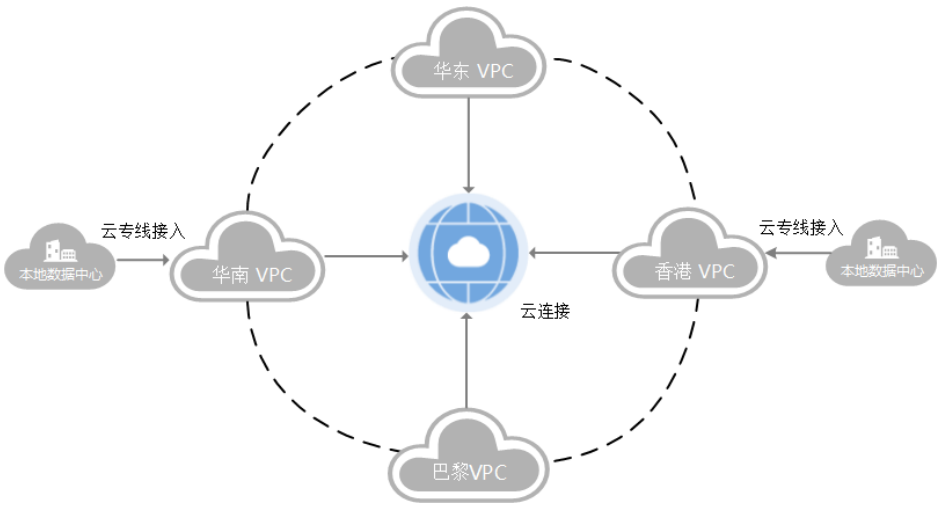
图 4-5 通过 VPN 实现不同区域 VPC 互连



- 云连接

  云连接（Cloud Connect）为用户提供一种能够快速构建跨区域VPC之间以及云上多VPC与云下多数据中心之间的高速、优质、稳定的网络能力，帮助用户打造一张具有企业级规模和通信能力的全球云上网络。

图 4-6 使用云连接实现不同区域 VPC 互连



5 公网访问

公网产品

公有云提供弹性公网IP（EIP）、NAT网关、弹性负载均衡（ELB）等方式连接公网。

- EIP

  EIP提供独立的公网IP资源，包括公网IP地址与公网出口带宽服务。可以与弹性云服务器、裸金属服务器、虚拟IP、弹性负载均衡、NAT网关等资源灵活地绑定及解绑。拥有多种灵活的计费方式，可以满足各种业务场景的需要。

- ELB

  ELB将访问流量自动分发到多台弹性云服务器，扩展应用系统对外的服务能力，实现更高水平的应用容错。为负载均衡器配置需要监听的端口信息以及弹性云服务器，通过监听器来检查后端弹性云服务器的运行状态，确保将请求发送到正常的弹性云服务器上，提高系统可用性。

- NAT网关

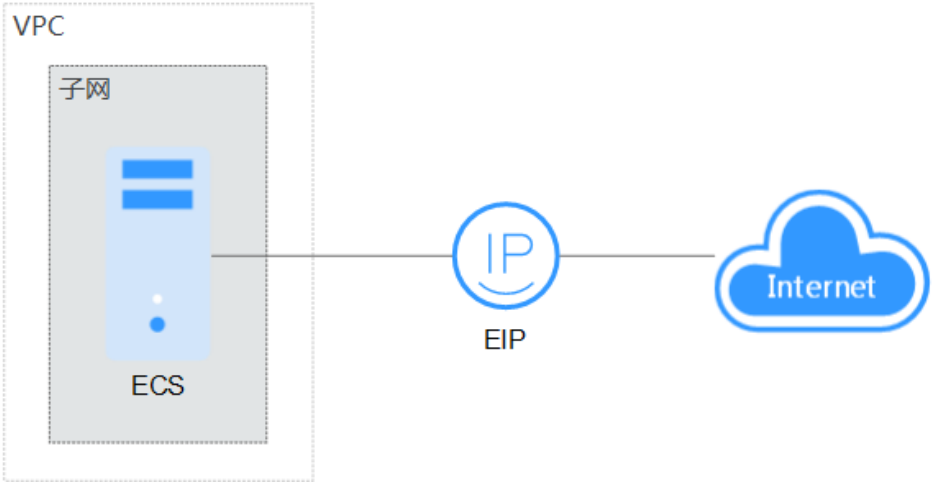
  NAT网关能够为VPC内的弹性云服务器提供SNAT和DNAT功能，通过灵活简易的配置，即可轻松构建VPC的公网出入口。

对外提供服务

- 单个ECS对外提供服务

  当您仅有单个应用服务，业务量较小时，您可申请一个EIP，绑定到ECS上，该ECS即可连接公网提供服务。

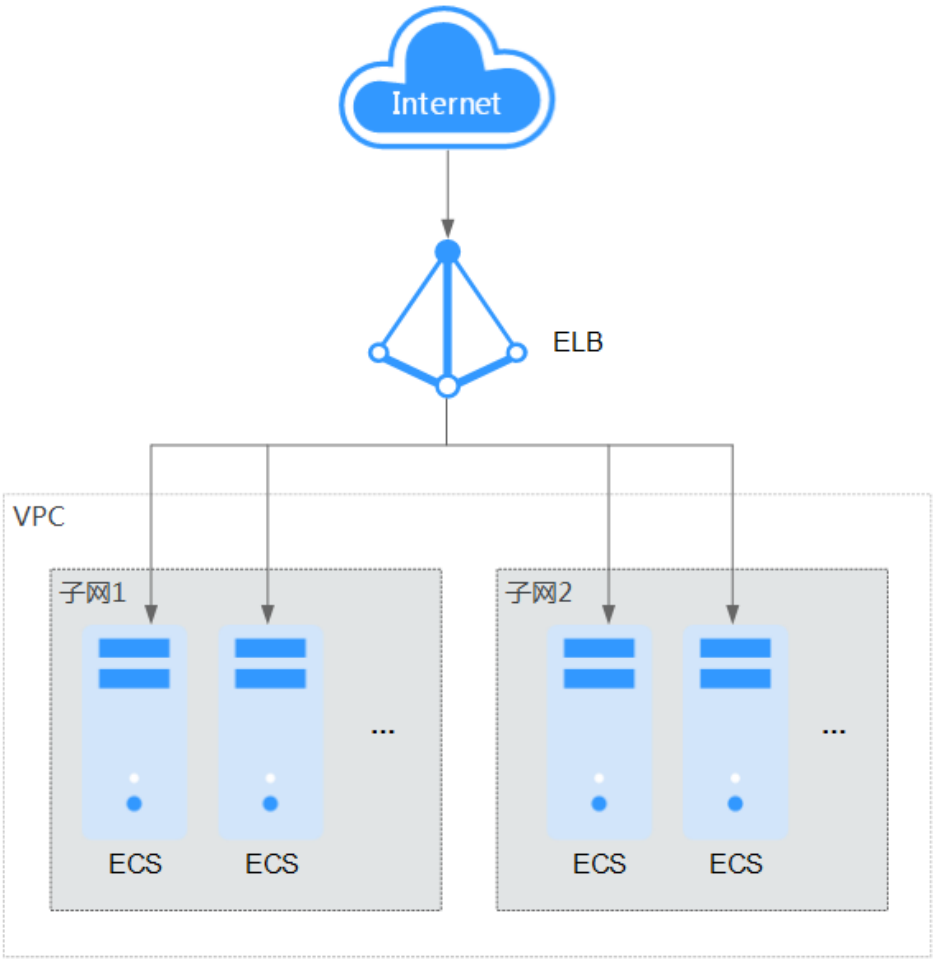
图 5-1 EIP



- 多个ECS负载均衡

  对于电商等高并发访问的场景，您可以通过ELB将访问流量均衡分发到多台弹性云服务器上，支撑海量用户访问。华为云ELB无缝集成了弹性伸缩服务，能够根据业务流量自动扩容，保证业务稳定可靠。

图 5-2 ELB

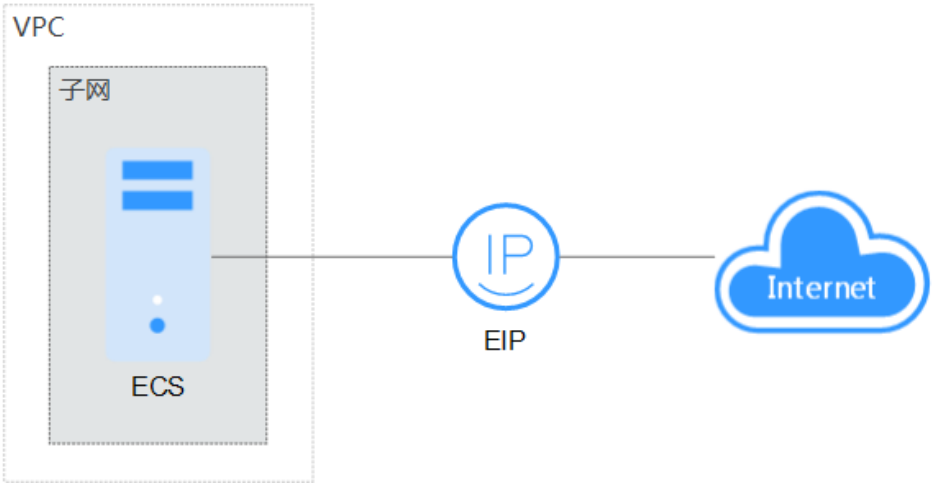


主动访问公网

- 单个ECS访问公网

  当您的某台ECS需要主动访问公网，可以为ECS绑定EIP，即可实现公网访问。华为云提供多种计费方式（按需等）供您选择，无需使用时支持灵活解绑。

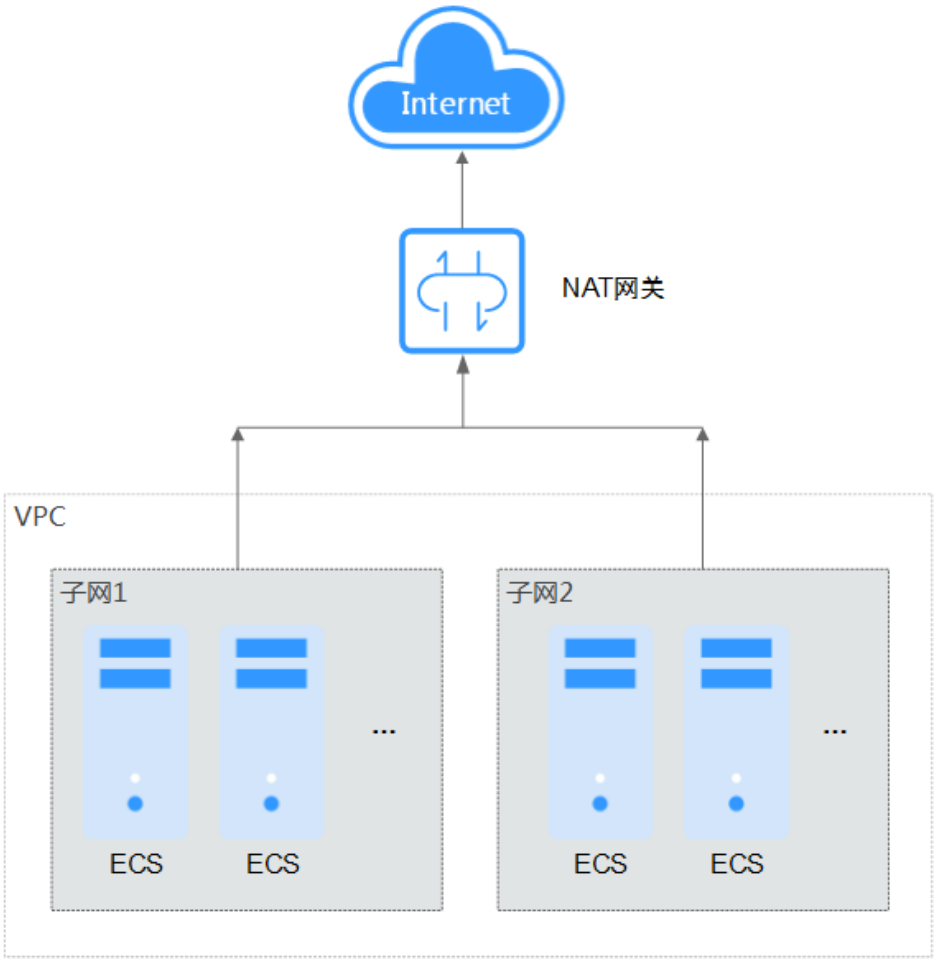
  图 5-3 EIP

- 多个ECS访问公网

  当您的VPC内ECS都有公网访问需求时，可以使用NAT网关服务，按子网配置SNAT规则，轻松构建VPC的公网出口。对比EIP访问公网，在未配置DNAT规则时，外部用户无法通过公网直接访问NAT网关的公网地址，保证了ECS的相对安全。

图 5-4 NAT 网关



6 节约公网成本

在您购买带宽产品之前一定要分析您业务系统的流量模型，以便选择适合的产品及计费模式。

独享带宽

如您需要保证单个弹性公网IP的带宽大小，建议您购买独享带宽。独享带宽只针对单个弹性公网IP进行限速，不受其他业务影响。

支持两种计费模式：

- 按带宽计费：针对流量使用较大且比较稳定的业务。
- 按流量计费：针对流量使用相对较小的业务，搭配共享流量包使用价格更优惠。

对于流量比较稳定，没有突发流量的系统可以考虑选择预付费的按带宽计费模式，可以比正常后付费按带宽计费享受更多价格优惠。

共享带宽

当您有大量业务在云上时，如果每个ECS单独使用一条独享带宽，则需要较多的带宽实例，并且总的带宽费用会较高，如果所有实例共用一条带宽，就可以节省企业的网络运营成本，同时方便运维统计。共享带宽是独立的带宽产品，支持将多个按需计费的弹性公网IP添加到共享带宽，对多个弹性公网IP进行集中限速。您可以将EIP绑定到ECS、NAT网关、ELB等产品，从而使这些产品使用共享带宽。

支持两种计费模式：

- 按带宽计费：如果您使用的弹性公网IP较多，并且错峰明显，使用共享带宽可以大幅节约成本。
- 按增强型95计费：如果您部署的业务经常有突发峰值，可以选择增强型95计费。既可以保证业务系统不受峰值带宽不够的影响，又可以避免带宽峰值设置过大带来的成本浪费。

共享流量包

共享流量包是公网流量的预付费套餐，价格比后付费流量更低，大大降低了公网流量成本。共享流量包购买后立即生效，自动抵扣按需计费（按流量计费）的EIP带宽产生的流量资费，使用简单，无需额外操作。

- 共享流量包适用哪些场景?

  对于按流量计费的带宽，启用共享流量包后，该带宽所产生的流量费用优先从共享流量包中进行抵扣。共享流量包全部使用完后，再按后付费流量进行结算。从节约成本的角度看，流量越大，节省的成本越多。

- 共享流量包使用说明
  - 只能抵扣同一区域产生的带宽流量，不支持跨区域抵扣。
  - 共享流量包包括动态和静态两种类型，分别抵扣全动态BGP和静态BGP产生的流量。
  - 共享流量包具有使用有效期（从购买开始计算1个自然月或1个自然年）。超过有效期后，没有使用完的流量无法继续使用。建议根据业务系统历史情况仔细评估需要多少共享流量包。
  - 共享流量包支持自动续费功能。如果您开通了自动续费功能，那么共享流量包到期前7天内，系统会尝试自动续费扣款，续费成功后，共享流量包中剩余的流量可以在新的有效期内继续使用。
  - 共享流量包全部使用完后，系统会自动按后付费流量进行结算，不会导致业务系统无法使用。

7 VPC 网络安全

7.1 使用 IP 地址组提升安全组规则管理效率

应用场景

IP地址组是一个或者多个IP地址的集合，您可以在配置安全组规则的时候使用IP地址组。如果您变更了IP地址组内的IP地址，则相当于直接变更了这些IP地址对应的安全组规则，免去逐条修改安全组规则的工作量。

通常情况下，针对金融，证券等企业，在规划云上组网业务时，对安全性要求较高，实例内的访问控制需要针对IP粒度进行配置。为了既能实现针对IP粒度的精细控制，又能确保安全组规则配置的简洁性，对于安全策略相同的IP网段和IP地址，建议您使用IP地址组降低管理安全组规则的工作量。关于IP地址组的更多信息，请参见**IP地址组简介**。

例如，某企业在云上部署在线办公系统，为企业内不同部门提供服务，并且按照业务安全等级，将实例划分到多个安全组内。这些实例需要被企业内多个部门同时访问，企业内用户IP地址数量众多，且经常会发生变动。

- 不使用IP地址组的情况下，工程师需要在多个安全组内，分别维护针对不同授权对象的多条安全组规则。一旦企业用户的IP地址发生变动，工程师需要逐个调整每个安全组内对应的规则。安全组数量和规则数量越多，管理工作量越大。
- 使用IP地址组的情况下，工程师可以将企业用户的IP地址添加到IP地址组内，并在安全组内添加针对该IP地址组的授权规则。当企业用户的IP地址发生变化时，工程师只需要在IP地址组内修改IP地址，那么IP地址组对应的安全组规则将会随之变更，无需修改每个安全组内的规则，降低了安全组管理的难度，提升效率。

方案架构

本示例中，用户根据不同的安全要求，将实例划分在三个安全组内，同时，这些实例均需要允许特定IP地址访问SSH(22)端口，为了方便维护，采用IP地址组方案。

1. 创建一个IP地址组，并添加待授权的IP地址。
2. 分别在三个安全组入方向中，添加授权IP地址组访问的规则。

表 7-1 安全组入方向规则说明

方向	策略	类型	协议端口	源地址
入方向	允许	IPv4	TCP: 22	IP地址组

3. 如果后续允许访问实例的IP地址有变化，此时需要在IP地址组内修改IP地址条目，对应的安全组规则会自动生效。

约束与限制

对于关联IP地址组的安全组，其中IP地址组相关的规则对某些类型的云服务器不生效，不支持的类型如下：

- 通用计算型（S1型、C1型、C2型 ）
- 内存优化型（M1型）
- 高性能计算型（H1型）
- 磁盘增强型（ D1型）
- GPU加速型（G1型、G2型）
- 超大内存型（E1型、E2型、ET2型）

资源规划

本示例中需要规划的IP地址组和安全组资源需要位于同一个区域内，详细说明如**表7-2**所示。以下资源规划详情仅为示例，您可以根据需要自行修改。

表 7-2 资源规划说明

资源类型	资源数量	说明
IP地址组	1	创建IP地址组，并添加指定IP地址。 ● 名称：ipGroup-A ● 最大条目数：请根据实际情况填写，本示例为20。 ● IP类型：请根据实际情况填写，本示例为IPv4。 ● IP地址条目： – 11.xx.xx.64/32 – 116.xx.xx.252/30 – 113.xx.xx.0/25 – 183.xx.xx.208/28
安全组	3	在3个安全组中，均需要添加授权IP地址组访问的规则，具体如**表7-3**所示。

表 7-3 安全组入方向规则说明

方向	策略	类型	协议端口	源地址
入方向	允许	IPv4	TCP: 22	ipGroup-A

操作步骤

步骤1 创建一个IP地址组，并添加指定IP地址。

具体操作请参见**创建IP地址组**。

步骤2 在3个安全组中，分别添加授权IP地址组访问的规则。

具体操作请参见**添加安全组规则**。

添加完成后，允许来自11.xx.xx.64/32、116.xx.xx.252/30、113.xx.xx.0/25、183.xx.xx.208/28的流量访问安全组内实例的SSH(22)端口，通常用于远程登录Linux云服务器。

步骤3 修改IP地址组内的IP地址条目。

如果添加安全组规则后，又需要为新增的IP地址添加授权规则，此时您需要在IP地址组内增加新的IP地址即可。比如，在IP地址组内增加网段117.xx.xx.0/25后，安全组规则自动生效，允许来自117.xx.xx.0/25的流量访问安全组内实例的SSH(22)端口。

具体操作请参见**管理IP地址组内的IP地址条目**。

----结束

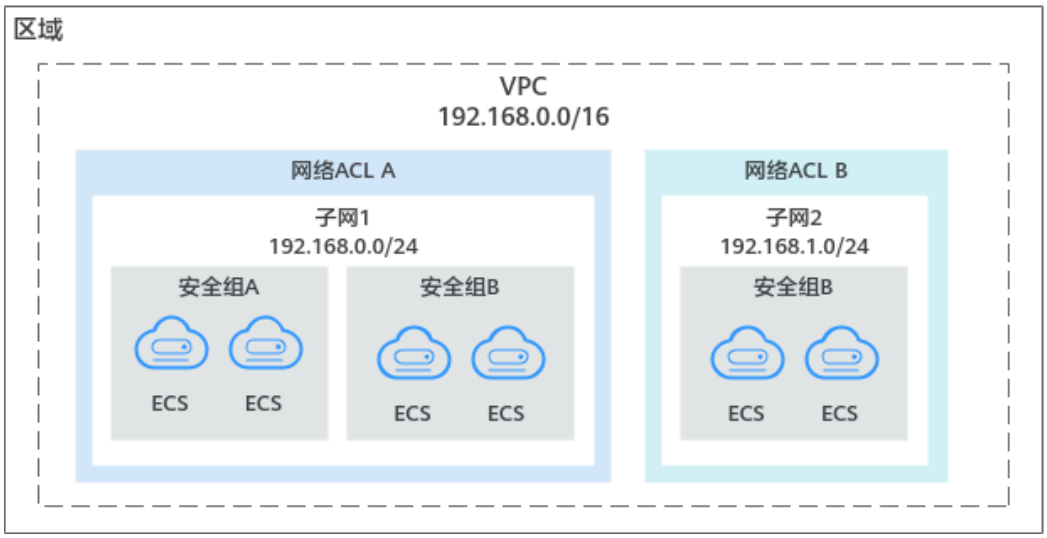
7.2 通过安全组和网络 ACL 实现 VPC 的访问控制

虚拟私有云VPC是您在云上的私有网络，通过配置安全组和网络ACL策略，可以保障VPC内部署的实例安全运行，比如弹性云服务器、数据库、云容器等。

- 安全组对实例进行防护，将实例加入安全组内后，该实例将会受到安全组的保护。
- 网络ACL对整个子网进行防护，将子网关联至网络ACL，则子网内的所有实例都会受到网络ACL保护。相比安全组，网络ACL的防护范围更大。

如**图7-1**所示，安全组A和安全组B可以保护其中ECS的网络安全，通过网络ACL A和网络ACL B，可以分别保护整个子网1和子网2的安全，双层防护提升安全保障。

图 7-1 安全组与网络 ACL



以下为您介绍一些常用的安全组和网络ACL的配置示例：

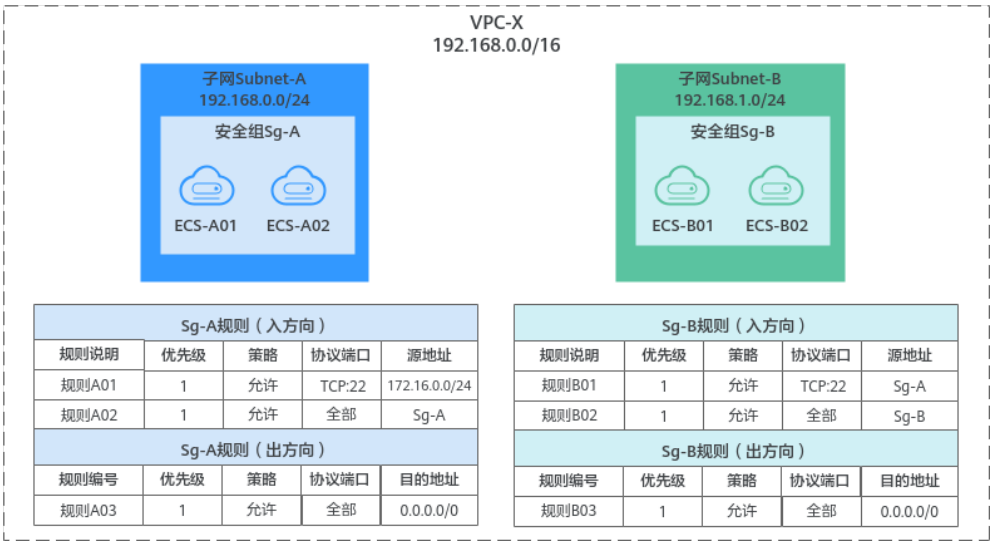
- 安全组：**控制外部指定IP地址或安全组对实例的访问**
- 安全组：**控制虚拟IP访问安全组内实例**
- 安全组：**控制对等连接两端VPC内的实例互访**
- 网络ACL：**控制外部对子网内实例的访问**
- 网络ACL：**控制不同子网内实例的互通和隔离**

控制外部指定 IP 地址或安全组对实例的访问

本示例安全组配置如**图7-2**所示，您可以通过设置安全组入方向规则，允许特定IP地址，或者其他安全组内的实例访问您的实例。

- 在安全组Sg-A的入方向中，添加规则A01，允许指定IP (172.16.0.0/24)访问安全组内实例的SSH(22)端口，用于远程登录安全组内的Linux云服务器。
- 在安全组Sg-B的入方向中，添加规则B01，允许其他安全组内的实例访问本安全组内实例的SSH(22)端口，即通过子网Subnet-A的ECS可远程登录Subnet-B内的ECS。

图 7-2 控制外部指定 IP 地址或安全组对实例的访问



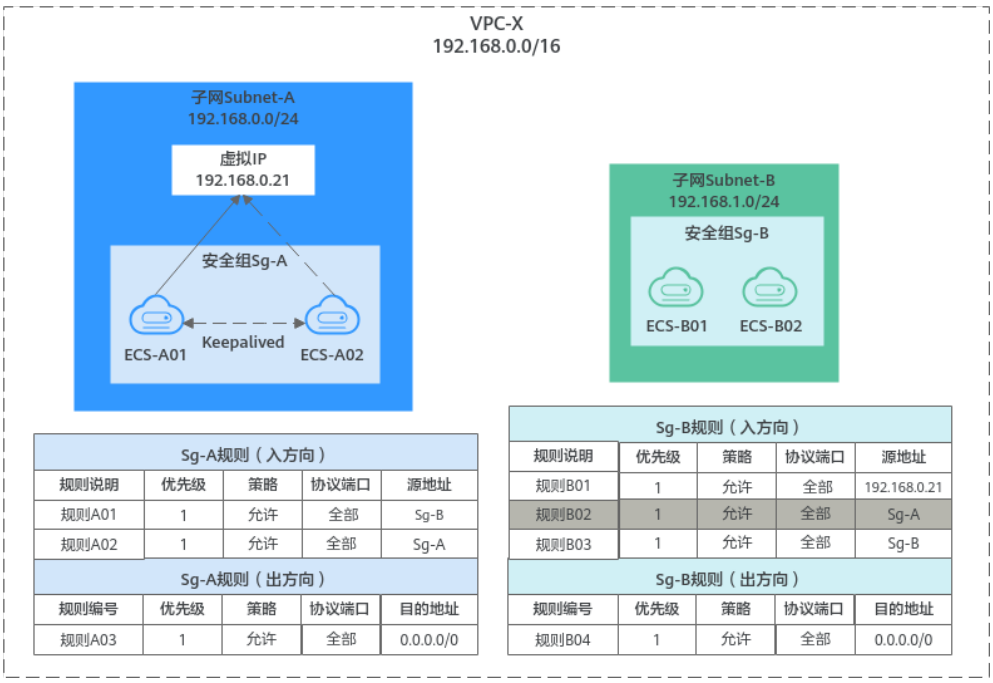
控制虚拟 IP 访问安全组内实例

本示例安全组配置如图7-3所示，您可以通过设置安全组入方向规则，允许虚拟IP，或者其他安全组内的实例访问您的实例。

- 在安全组Sg-A的入方向中，添加规则A01，允许Sg-B内的实例使用任何协议和端口，通过私有IP地址访问Sg-A内的实例。
- 在安全组Sg-B的入方向中，添加规则B01，允许虚拟IP(192.168.0.21)使用任何协议和端口访问Sg-B内的实例。当前组网中，您还可以将源地址设置成子网Subnet-A的网段192.168.0.0/24。

  规则B02仅能允许Sg-A内实例通过私有IP访问Sg-B内的实例，无法放通虚拟IP访问。

图 7-3 控制虚拟 IP 访问安全组内实例

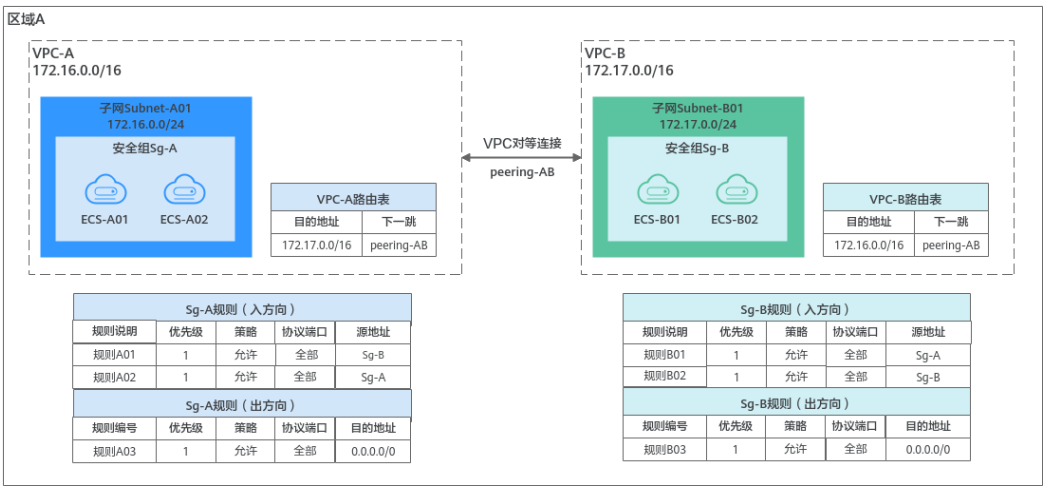


控制对等连接两端 VPC 内的实例互访

本示例安全组配置如图7-4所示。您可以通过设置安全组入方向规则，实现对等连接两端的ECS网络互通。

- 在安全组Sg-A中，添加规则A01，允许来自Sg-B内实例的流量访问Sg-A内的实例，源地址为安全组Sg-B。
- 在安全组Sg-B中，添加规则B01，允许来自Sg-A内实例的流量访问Sg-B内的实例，源地址为安全组Sg-A。

图 7-4 控制对等连接两端 VPC 内的实例互访



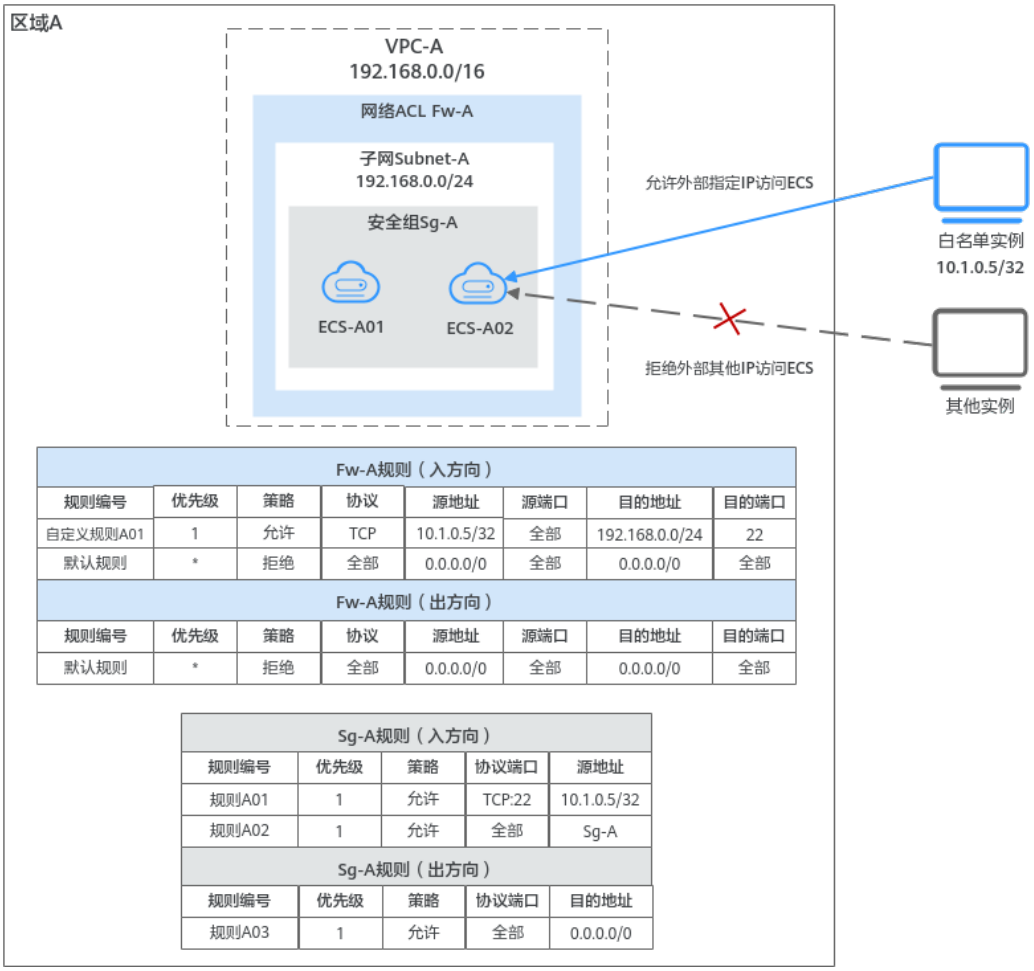
控制外部对子网内实例的访问

网络ACL可以控制流入/流出子网的流量，流量优先匹配网络ACL的规则，然后匹配安全组规则。

本示例如图7-5所示，子网Subnet-A内的两个业务实例ECS-A01和ECS-A02网络互通，并允许白名单实例远程登录业务实例，白名单实例的IP地址为10.1.0.5/32。白名单实例可能是VPC-A的其他子网或者其他VPC内的实例，也可以是本地计算机，可远程连接业务实例执行运维操作。因此，网络ACL和安全组规则需要放通白名单实例的流量，拦截来自其他网络的流量，规则配置如下：

- 网络ACL规则：
  - 入方向：自定义规则A01允许白名单实例，通过SSH远程登录子网Subnet-A内的实例。默认规则拒绝其他网络流量流入子网。
  - 出方向：网络ACL是有状态的，允许入站请求的响应流量流出，因此不用额外添加规则放通白名单实例的响应流量。默认规则拒绝其他网络流量流出子网。
- 安全组规则：
  - 入方向：规则A01允许白名单实例，通过SSH远程登录子网Subnet-A内的实例。规则A02允许安全组内实例互通。其他流量无法流入安全组内实例。
  - 出方向：规则A03允许所有流量从安全组内实例流出。

图 7-5 控制外部对子网内实例的访问



Fw-A规则（入方向）							
规则编号	优先级	策略	协议	源地址	源端口	目的地址	目的端口
自定义规则A01	1	允许	TCP	10.1.0.5/32	全部	192.168.0.0/24	22
默认规则	*	拒绝	全部	0.0.0.0/0	全部	0.0.0.0/0	全部
Fw-A规则（出方向）							
规则编号	优先级	策略	协议	源地址	源端口	目的地址	目的端口
默认规则	*	拒绝	全部	0.0.0.0/0	全部	0.0.0.0/0	全部

Sg-A规则（入方向）				
规则编号	优先级	策略	协议端口	源地址
规则A01	1	允许	TCP:22	10.1.0.5/32
规则A02	1	允许	全部	Sg-A
Sg-A规则（出方向）				
规则编号	优先级	策略	协议端口	目的地址
规则A03	1	允许	全部	0.0.0.0/0

控制不同子网内实例的互通和隔离

本示例中，VPC-X内有两个子网Subnet-X01和Subnet-X02，ECS-01和ECS-02属于Subnet-X01，ECS-03属于Subnet-X02。三台ECS的网络通信需求如下：

- ECS-02和ECS-03网络互通
- ECS-01和ECS-03网络隔离

为了实现以上网络通信需求，本示例的安全组和网络ACL配置如下：

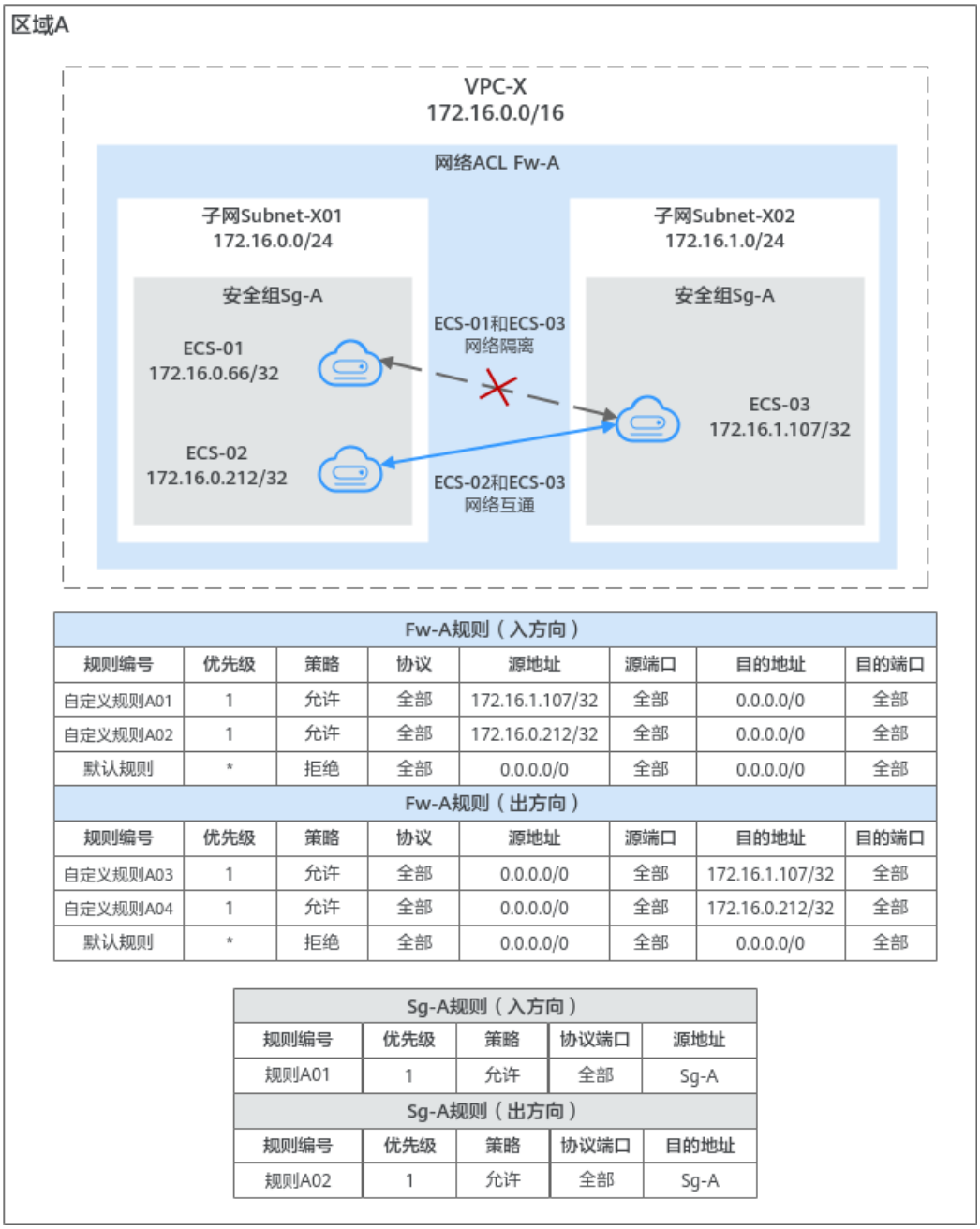
1. 三台ECS属于同一个安全组Sg-A，在Sg-A中添加入方向和出方向规则，确保安全组内实例网络互通。

   此时子网还未关联网络ACL，安全组规则配置完成后，ECS-01、ECS-02均可以和ECS-03进行通信。
2. 将两个子网均关联至网络ACL Fw-A。

   当Fw-A中只有默认规则时，同一个子网内实例网络互通，不同子网内实例网络隔离。此时ECS-01和ECS-02网络互通，ECS-01和ECS-03网络隔离、ECS-02和ECS-03网络隔离。
3. 在网络ACL Fw-A中添加自定义规则，放通ECS-02和ECS-03之间的网络。
   - 自定义规则A01：允许来自ECS-03的流量流入子网。
   - 自定义规则A02：允许来自ECS-02的流量流入子网。

- 自定义规则A03：允许访问ECS-03的流量流出子网。
- 自定义规则A04：允许访问ECS-02的流量流出子网。

图 7-6 控制不同子网内实例的互通和隔离



Fw-A规则（入方向）							
规则编号	优先级	策略	协议	源地址	源端口	目的地址	目的端口
自定义规则A01	1	允许	全部	172.16.1.107/32	全部	0.0.0.0/0	全部
自定义规则A02	1	允许	全部	172.16.0.212/32	全部	0.0.0.0/0	全部
默认规则	*	拒绝	全部	0.0.0.0/0	全部	0.0.0.0/0	全部
Fw-A规则（出方向）							
规则编号	优先级	策略	协议	源地址	源端口	目的地址	目的端口
自定义规则A03	1	允许	全部	0.0.0.0/0	全部	172.16.1.107/32	全部
自定义规则A04	1	允许	全部	0.0.0.0/0	全部	172.16.0.212/32	全部
默认规则	*	拒绝	全部	0.0.0.0/0	全部	0.0.0.0/0	全部

Sg-A规则（入方向）				
规则编号	优先级	策略	协议端口	源地址
规则A01	1	允许	全部	Sg-A
Sg-A规则（出方向）				
规则编号	优先级	策略	协议端口	目的地址
规则A02	1	允许	全部	Sg-A

7.3 通过对等连接和第三方防火墙实现多 VPC 互访流量清洗

应用场景

虚拟私有云支持用户自主配置和管理虚拟网络环境，您可以在VPC中使用安全组及网络ACL来进行网络访问控制，也可以使用第三方防火墙软件，对云上的业务进行灵活的安全控制。

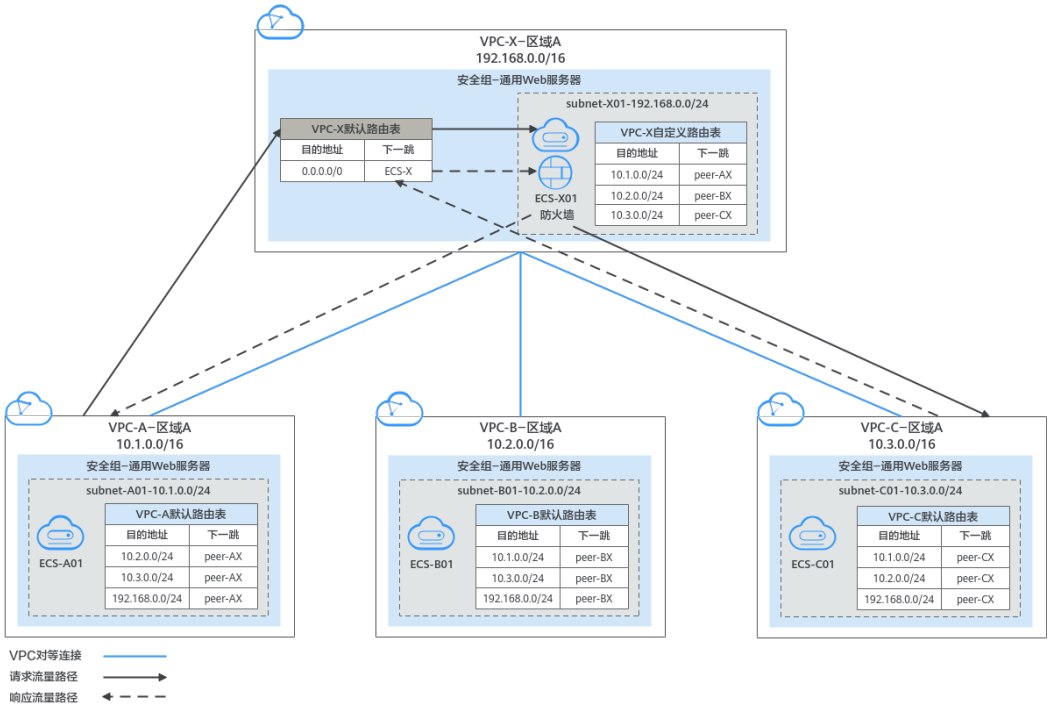
本文为您介绍通过防火墙软件实现VPC内流量安全管控的需求，首先基于VPC对等连接实现多个VPC网络互通，然后VPC之间互访流量通过防火墙软件过滤清洗。

方案架构

本示例中vpc-A、vpc-B、vpc-C为业务所在的VPC，vpc-X为防火墙所在的VPC，这些VPC通过对等连接实现网络互通。vpc-A、vpc-B、vpc-C之间互通的流量均需要经过vpc-X上的防火墙。根据默认路由表配置，所有vpc-X的入方向流量均引入防火墙，通过防火墙清洗后的流量根据自定义路由表的目的地址送往指定业务VPC。

在图7-7中，以ecs-A01访问ecs-C01为例，您可以清晰的看到流量的请求路径和响应路径。

图 7-7 云上 VPC 互访使用第三方防火墙组网规划



资源规划说明

本示例中需要创建虚拟私有云VPC、弹性云服务器ECS以及VPC对等连接，资源规划总体说明请参见表7-4。

说明

以下资源规划详情仅为示例，供您参考，您需要根据实际业务情况规划资源。

表 7-4 云上 VPC 互访使用第三方防火墙资源规划总体说明

资源	说明
虚拟私有云 VPC	VPC的资源规划详情如表7-5所示。 本示例中共有4个VPC，包括业务所在VPC和防火墙所在的VPC。这些VPC位于同一个区域内，且这些VPC的子网网段不重叠。 ● vpc-A、vpc-B、vpc-C为业务VPC，vpc-X为防火墙VPC，这些VPC通过对等连接实现网络互通。 ● vpc-A、vpc-B、vpc-C、vpc-X各有一个子网。 ● vpc-A、vpc-B、vpc-C各有一个默认路由表，子网关联VPC默认路由表。 ● vpc-X有两个路由表，一个系统自带的默认路由表，一个用户创建的自定义路由表，vpc-X的子网关联自定义路由表。 默认路由表控制vpc-X的入方向流量，自定义路由表控制vpc-X的出方向流量。 **须知** 需要通过对等连接通信的VPC的子网网段不能重叠，否则对等连接不会生效，更多详情请参见无效的VPC对等连接配置。
弹性云服务器 ECS	ECS的资源规划详情如表7-6所示。 本示例中共有4个ECS，这些ECS分别位于不同的VPC内，这些ECS如果位于不同的安全组，需要在安全组中添加规则放通对端安全组的网络。
VPC对等连接	VPC对等连接的资源规划详情如表7-7所示。 本示例中共3个对等连接，网络连通需求如下： ● peer-AX：连通vpc-A和vpc-X的网络。 ● peer-BX：连通vpc-B和vpc-X的网络。 ● peer-CX：连通vpc-C和vpc-X的网络。 由于VPC对等连接具有传递性，通过路由配置，vpc-A、vpc-B以及vpc-C之间可以通过vpc-X进行网络通信。

表 7-5 VPC 资源规划详情

VPC名称	VPC网段	子网名称	子网网段	关联路由表	子网作用
vpc-A	10.1.0.0/16	subnet-A01	10.1.0.0/24	默认路由表	部署业务的子网
vpc-B	10.2.0.0/16	subnet-B01	10.2.0.0/24	默认路由表	部署业务的子网
vpc-C	10.3.0.0/16	subnet-C01	10.3.0.0/24	默认路由表	部署业务的子网
vpc-X	192.168.0.0/16	subnet-X01	192.168.0.0/24	自定义路由表	部署防火墙的子网

表 7-6 ECS 资源规划详情

ECS名称	VPC名称	子网名称	私有IP地址	镜像	安全组	ECS作用
ecs-A01	vpc-A	subnet-A01	10.1.0.139	公共镜像：CentOS 8.2 64bit	sg-demo：通用Web服务器	部署业务的云服务器
ecs-B01	vpc-B	subnet-B01	10.2.0.93			部署业务的云服务器
ecs-C01	vpc-C	subnet-C01	10.3.0.220			部署业务的云服务器
ecs-X01	vpc-X	subnet-X01	192.168.0.5			部署防火墙的云服务器

表 7-7 VPC 对等连接资源规划详情

VPC对等连接名称	本端VPC	对端VPC
peer-AX	vpc-A	vpc-X
peer-BX	vpc-B	vpc-X
peer-CX	vpc-C	vpc-X

组网规划说明

本示例中需要在VPC路由表中配置路由，实现VPC之间的互通以及通过防火墙的流量清洗、组网规划总体说明请参见**表7-8**。

说明

以下路由规划详情仅为示例，供您参考，您需要根据实际业务情况规划路由。

表 7-8 云上 VPC 互访使用第三方防火墙组网规划总体说明

路由表	说明
业务所在VPC	vpc-A、vpc-B、vpc-C为业务VPC，路由表的规划详情如**表7-9**所示。 在vpc-A、vpc-B、vpc-C的默认路由表中，分别添加指向其他VPC子网，下一跳为对等连接的路由，实现不同VPC之间的网络互通。

路由表	说明
防火墙所在VPC	vpc-X为防火墙VPC，路由表的规划详情如表7-10所示。 1. 在vpc-X的默认路由表中，根据您防火墙部署方案分为以下情况： • 防火墙部署在一台ECS上，则添加目的地址为默认网段（0.0.0.0/0），下一跳为ecs-X01的路由，将流量引入防火墙所在的云服务器。 • 防火墙部署在两台ECS上，对外通过同一个虚拟IP通信，当主ECS发生故障无法对外提供服务时，动态将虚拟IP切换到备ECS，继续对外提供服务。此场景下，则添加目的地址为默认网段（0.0.0.0/0），下一跳为虚拟IP的路由，将流量进入虚拟IP，由虚拟IP将流量引入防火墙所在的云服务器。 本文以防火墙部署在一台ECS上为例，vpc-A、vpc-B、vpc-C互访的流量，都需要经过vpc-X，然后通过该条路由，将流量引入防火墙中进行清洗过滤。 2. 在vpc-X的自定义路由表中，添加目的地址为业务VPC子网网段（vpc-A、vpc-B、vpc-C子网），下一跳为对等连接的路由，将清洗后的流量引入业务VPC。

表 7-9 业务 VPC 路由表规划

VPC名称	VPC路由表	目的地址	下一跳类型	下一跳	路由类型	路由作用
vpc-A	默认路由表：rtb-vpc-A	10.2.0.0/24	对等连接	peer-AX	自定义	• 目的地址指向vpc-B的子网subnet-B01 • 连通子网subnet-A01和subnet-B01
		10.3.0.0/24	对等连接	peer-AX	自定义	• 目的地址指向vpc-C的子网subnet-C01 • 连通子网subnet-A01和subnet-C01
		192.168.0.0/24	对等连接	peer-AX	自定义	• 目的地址指向vpc-X的子网subnet-X01 • 连通子网subnet-A01和subnet-X01

VPC名称	VPC路由表	目的地址	下一跳类型	下一跳	路由类型	路由作用
vpc-B	默认路由表：rtb-vpc-B	10.1.0.0/24	对等连接	peer-BX	自定义	• 目的地址指向vpc-A的子网subnet-A01 • 连通子网subnet-A01和subnet-B01
		10.3.0.0/24	对等连接	peer-BX	自定义	• 目的地址指向vpc-C的子网subnet-C01 • 连通子网subnet-B01和subnet-C01
		192.168.0.0/24	对等连接	peer-BX	自定义	• 目的地址指向vpc-X的子网subnet-X01 • 连通子网subnet-B01和subnet-X01
vpc-C	默认路由表：rtb-vpc-C	10.1.0.0/24	对等连接	peer-CX	自定义	• 目的地址指向vpc-A的子网subnet-A01 • 连通子网subnet-A01和subnet-C01
		10.2.0.0/24	对等连接	peer-CX	自定义	• 目的地址指向vpc-B的子网subnet-B01 • 连通子网subnet-B01和subnet-C01
		192.168.0.0/24	对等连接	peer-CX	自定义	• 目的地址指向vpc-X的子网subnet-X01 • 连通子网subnet-C01和subnet-X01

表 7-10 防火墙 VPC 路由表规划

VPC名称	VPC路由表	目的地址	下一跳类型	下一跳	路由类型	路由作用
vpc-X	默认路由表：rtb-vpc-X	0.0.0.0/0	服务器实例	ECS-X	自定义	• 目的地址指向部署防火墙的ecs-X • 将vpc-X入方向的流量引入防火墙 本文以防火墙部署在一台ECS上为例，如果您的防火墙同时部署在多台ECS上，对外通过虚拟IP通信，则路由下一跳选择虚拟IP。
	自定义路由表：rtb-vpc-custom-X	10.1.0.0/24	对等连接	peer-AX	自定义	• 目的地址指向vpc-A的子网subnet-A01 • 连通子网subnet-A01和subnet-X01
		10.2.0.0/24	对等连接	peer-BX	自定义	• 目的地址指向vpc-B的子网subnet-B01 • 连通子网subnet-B01和subnet-X01
		10.3.0.0/24	对等连接	peer-CX	自定义	• 目的地址指向vpc-C的子网subnet-C01 • 连通子网subnet-C01和subnet-X01

约束与限制

- VPC对等连接只能实现同区域VPC的网络互通，因此请确保您的VPC位于同一个区域内。
- 需要通过VPC对等连接通信的VPC的子网网段不能重叠，否则对等连接不会生效，更多详情请参见**无效的VPC对等连接配置**。
- 第三方防火墙部署的ECS所在的子网需要关联自定义路由表 ，请确保您资源所在的区域支持自定义路由表功能。

  如果在网络控制台的左侧子栏目看到独立的“路由表”选项，表示支持自定义路由表功能。

图 7-8 支持定义路路由

操作步骤

步骤1 在区域A内，创建4个VPC及其子网。

具体方法请参见**创建虚拟私有云和子网**。

本示例中的VPC和子网资源规划详情请参见**表7-5**。

步骤2 创建vpc-X内的自定义路由表，并将subnet-X01关联至自定义路由表。

1. 在vpc-X内，创建自定义路由表。
   具体方法请参见**创建自定义路由表**。
2. 将子网subnet-X01的关联至**步骤2.1**创建的自定义路由表。
   子网创建完成后，自动关联VPC默认路由表，因此当前子网subnet-X01关联的是vpc-X的默认路由表，需要更换为**步骤2.1**创建的自定义路由表。
   具体方法请参见**更换子网关联的路由表**。

步骤3 创建四个ECS，分别属于不同的VPC内。

创建ECS，具体方法请参见**创建弹性云服务器**。

步骤4 配置ecs-X的网卡，并安装第三方防火墙软件。

1. 关闭ecs-X的网卡“源/目的检查”。
   a. 在ECS列表中，单击目标ECS的名称。
      进入ECS详情页。
   b. 选择“弹性网卡”页签，并单击 展开ECS的网卡详情区域，可以查看“源/目的检查”功能。
      如**图7-9**所示，表示“源/目的检查”功能已关闭。

图 7-9 关闭网卡的“源/目的检查”功能

2. 在ecs-X中安装第三方防火墙。

   您可以自行安装或者通过**华为云商店**购买第三方防火墙。

步骤5 （可选）为云服务器配置虚拟IP。

此步骤为可选：您可以在vpc-X中创建主备服务器，并绑定同一虚拟IP，当主服务器发生故障无法对外提供服务时，动态将虚拟IP切换到备服务器，继续对外提供服务。如果部署第三方防火墙的弹性云服务器不需要主备，此步骤不需要执行。

1. 在vpc-X的子网内，创建虚拟IP。

   具体方法请参见**申请虚拟IP地址**。

2. 将虚拟IP绑定到部署防火墙的主备ECS上。

   具体方法请参见**虚拟IP绑定云服务器**。

步骤6 创建3个VPC对等连接，并配置路由。

1. 创建3个VPC对等连接。
   - 如果您的VPC在同一个账号内，具体方法请参见**创建同一账户下的对等连接**，您只需要执行该章节的“创建VPC对等连接”小节。
   - 如果您的VPC在不同一个账号内，具体方法请参见**创建不同账户下的对等连接**，您需要执行该章节的“创建VPC对等连接”小节和“接受对等连接”小节。

   本示例中的VPC对等连接资源规划详情请参见**表7-7**。

2. 在3个业务VPC的默认路由表中，添加指向其他3个VPC，下一跳为对等连接的路由。

   具体方法请参见**添加自定义路由**。

   本示例中，分别在vpc-A、vpc-B、vpc-C的路由表中，添加**表7-9**中规划的路由。

3. 在防火墙VPC的默认路由表和自定义路由表中，分别配置路由。

   具体方法请参见**添加自定义路由**。

本示例中，分别在vpc-X的默认路由表和自定义路由表中，添加**表7-10**中规划的路由。

步骤7 登录ECS，验证防火墙是否生效。

弹性云服务器有多种登录方法，具体请参见**登录弹性云服务器**。

本示例是通过管理控制台远程登录（VNC方式）。

1. 登录ecs-A01，验证vpc-A与vpc-B网络互通情况。

   ping *ecs-B01的私有IP地址*

   命令示例：

   ping 10.2.0.93

   回显类似如下信息，表示网络互通配置成功。

   ```
   [root@ecs-A01 ~]# ping 10.2.0.93
   PING 10.2.0.93 (10.2.0.93) 56(84) bytes of data.
   64 bytes from 10.2.0.93: icmp_seq=1 ttl=64 time=0.849 ms
   64 bytes from 10.2.0.93: icmp_seq=2 ttl=64 time=0.455 ms
   64 bytes from 10.2.0.93: icmp_seq=3 ttl=64 time=0.385 ms
   64 bytes from 10.2.0.93: icmp_seq=4 ttl=64 time=0.372 ms
   ...
   --- 10.2.0.93 ping statistics ---
   ```

2. 不要中断**步骤7.1**，登录ecs-X01，验证vpc-A到vpc-B的流量是否通过ecs-X01。
3. 在ecs-X01上，执行以下命令，检查eth0网卡的流量变化。

   至少连续执行两次命令，检查RX packets和TX packets是否变化。

   ifconfig eth0

   流量变化说明，表示流量通过ecs-X01，流量被防火墙过滤。

   ```
   [root@ecs-X01 ~]# ifconfig eth0
   eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
         inet 192.168.0.5  netmask 255.255.255.0  broadcast 192.168.0.255
         inet6 fe80::f816:3eff:feb6:a632  prefixlen 64  scopeid 0x20<link>
         ether fa:16:3e:b6:a6:32  txqueuelen 1000  (Ethernet)
         RX packets 726222  bytes 252738526 (241.0 MiB)
         RX errors 0  dropped 0  overruns 0  frame 0
         TX packets 672597  bytes 305616882 (291.4 MiB)
         TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

   [root@ecs-X01 ~]# ifconfig eth0
   eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
         inet 192.168.0.5  netmask 255.255.255.0  broadcast 192.168.0.255
         inet6 fe80::f816:3eff:feb6:a632  prefixlen 64  scopeid 0x20<link>
         ether fa:16:3e:b6:a6:32  txqueuelen 1000  (Ethernet)
         RX packets 726260  bytes 252748508 (241.0 MiB)
         RX errors 0  dropped 0  overruns 0  frame 0
         TX packets 672633  bytes 305631756 (291.4 MiB)
         TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
   ```

4. 参考**步骤7.1**~**步骤7.3**，检查其他VPC之间的通信情况。

----结束

7.4 通过第三方防火墙实现 VPC 和云下数据中心互访流量清洗

操作场景

用户IDC数据中心和华为云通过云专线（DC）或虚拟专用网络（VPN）通信成功，在华为云的内网上使用第三方虚拟化防火墙，使得云上云下的业务流量经过自定义的第三方防火墙，对云上的业务进行灵活的安全控制。

本文以用户同区域的多VPC与本地IDC连通为例，介绍混合云使用第三方防火墙的应用场景。

方案优势

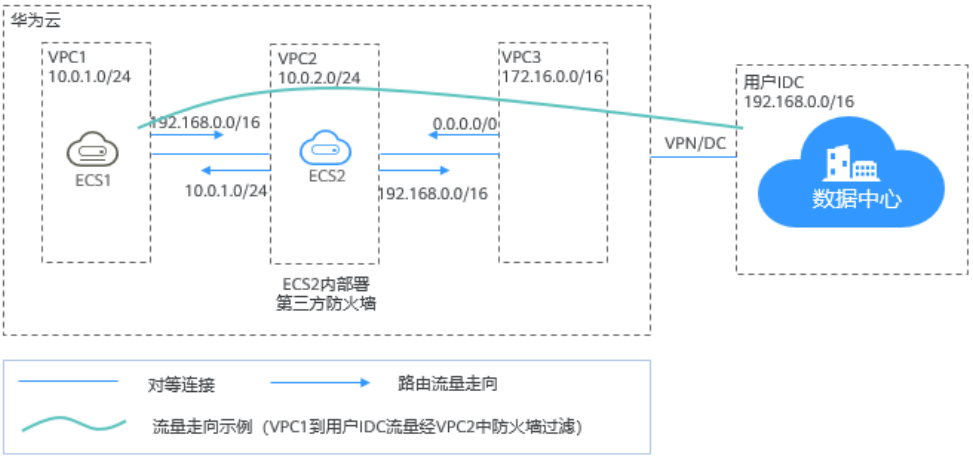
- 支持用户的第三方防火墙。
- 用户云上云下流量经过第三方防火墙。
- 支持用户自定义的更加灵活的安全策略。

典型拓扑

假设用户业务部署在VPC1、VPC2、VPC3及本地IDC中，并且需要在云上使用第三方虚拟化防火墙。用户可以将第三方虚拟化防火墙配置在VPC2的弹性云服务器中，使用对等连接及路由规则将VPC间进行连通。同时，在VPC3中创建云专线，使云上VPC与云下IDC实现连通。

实现方式如下：

图 7-10 场景示意



前提条件

VPC1与VPC2，VPC3子网网段不能重叠，否则对等连接无法通信成功。

配置步骤

步骤1 创建VPC

创建VPC1，VPC2，VPC3。

具体操作请参见**创建虚拟私有云和子网**。

说明

创建的VPC1，VPC2，VPC3网段不能重叠。例如VPC1：10.0.1.0/24；VPC2：10.0.2.0/24；VPC3：172.16.0.0/16

步骤2 创建弹性云服务器

1. 创建ECS1，ECS2，分别属于VPC1的子网，VPC2的子网。
   具体操作请参见章节**创建弹性云服务器**。

   说明

   ECS2的网卡要关闭源/目的检查。
2. 在弹性云服务器ECS2中部署第三方防火墙。
   参考防火墙软件：**https://marketplace.huaweicloud.com/all/?q=JemYsueBq-WimSU**

步骤3 创建对等连接

VPC1和VPC2，VPC2和VPC3，分别创建对等连接，实现VPC间的连通。

创建对等连接时，先不配置本端和对端的路由规则，具体配置路由规则参见**配置路由规则**。

具体操作请参见**创建对等连接**。

步骤4 创建子网路由表

创建自定义路由表，关联VPC2的子网，控制VPC2的子网的出流量走向。

具体操作请参见**创建自定义路由表**。

步骤5 创建虚拟IP并绑定弹性云服务器（可选）

您可以在VPC2中创建主备服务器，并绑定同一虚拟IP，当主服务器发生故障无法对外提供服务时，动态将虚拟IP切换到备服务器，继续对外提供服务。如果不需要备用服务器，此步骤可以省略。

1. 在VPC2的子网下创建虚拟IP。
   具体操作请参见**申请虚拟IP地址**。
2. 将创建的虚拟IP绑定到弹性云服务器ECS2上。
   具体操作请参考**虚拟IP绑定云服务器**。

步骤6 创建云专线

使用VPC3创建专线，使云上VPC与云下IDC实现连通。具体操作参见**创建云专线**。

步骤7 配置路由规则

通过配置路由规则将指向目的地址的流量转发到指定的下一跳地址。

1. 修改VPC1的默认路由表，增加一条路由规则：

   VPC1 > 用户IDC，目的地址：用户IDC的CIDR，下一跳：VPC1与VPC2的对等连接。

2. 修改VPC2的默认路由表，增加一条路由规则：

   目的地址：0.0.0.0/0，下一跳：ECS2。

   如果涉及主备部署，创建了虚拟IP的情况下，此处下一跳是虚拟IP的地址。

3. 修改VPC2的子网路由表，增加两条规则：

   a. VPC2 > VPC1，目的地址：VPC1的CIDR，下一跳：VPC1与VPC2的对等连接。

   b. VPC2 > 用户IDC，目的地址：用户IDC的CIDR，下一跳：VPC2与VPC3的对等连接。

4. 修改VPC3的默认路由表，增加一条路由规则：

   目的地址：0.0.0.0/0，下一跳：VPC2和VPC3的对等连接。

   由于上述的**创建云专线**创建了专线，此处有一条系统自动下发的到专线的路由

----结束

配置验证

登录弹性云服务器ECS1访问用户IDC，在ECS2中可以收到ECS1发给用户IDC的报文，报文经过ECS2中的防火墙，被防火墙规则过滤。

8 基于华为云弹性云服务器自建容器并实现通信

操作场景

在不使用华为云容器产品的情况下，支持用户在华为云弹性云服务器中部署容器，并实现同一个子网中不同弹性云服务器内的容器相互通信。

方案优势

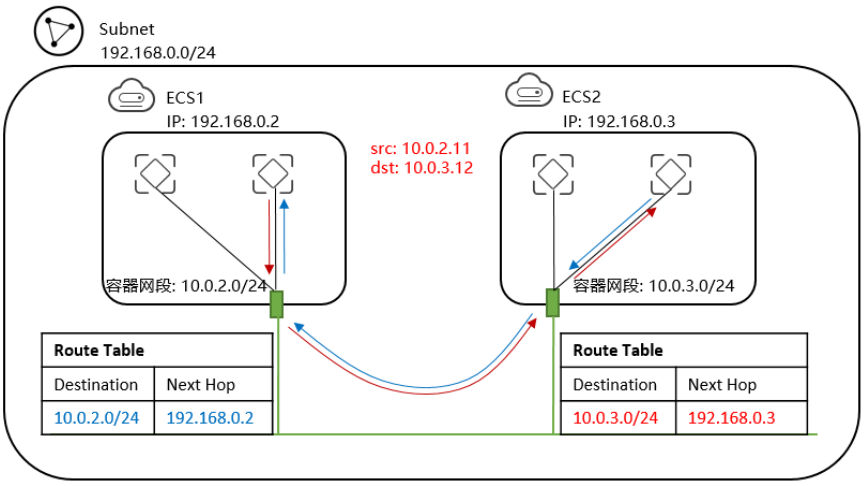
- 云服务器内部署容器，容器地址非VPC网络地址，通过VPC路由方式转发。
- 只需简单配置路由表，就可实现容器网络的互通，灵活方便。

典型拓扑

此场景下对网络拓扑有如下要求：

- 弹性云服务器在同一子网内。如图中VPC子网网段为192.168.0.0/24，弹性云服务器的IP地址为192.168.0.2和192.168.0.3。
- 容器网段与VPC子网不在一个网段，同一台弹性云服务器内的容器在同一个网段，不同弹性云服务器内容器的网段不同。如图中ECS1中容器网段为10.0.2.0/24，ECS2中容器网段为10.0.3.0/24。
- 发送给容器的数据包下一跳为容器所在弹性云服务器。如图中发送给10.0.2.0/24网段的数据包下一跳为192.168.0.2，发送给10.0.3.0/24网段的数据包下一跳为192.168.0.3。

图 8-1 网络拓扑



配置步骤

步骤1 创建VPC及VPC网段。

具体操作请参见**创建虚拟私有云和子网**。

步骤2 创建弹性云服务器。

具体操作请参见**创建弹性云服务器**。

创建完成后在弹性云服务器网卡上取消源地址校验，如**图8-2**所示。

图 8-2 取消源地址校验

步骤3 在弹性云服务器上部署容器。

您可以使用Docker CE完成容器的部署，详细操作步骤，请参考第三方软件的帮助文档，本文不做详细说明。

说明

同一台ECS内的容器需要在同一个网段，且不同ECS内容器网段不能重叠。

步骤4 添加VPC路由表信息。

在VPC路由表中添加路由信息。让发送给10.0.2.0/24网段的数据包下一跳为192.168.0.2，发送给10.0.3.0/24网段的数据包下一跳为192.168.0.3，也就是让发送给容器的数据包下一跳都为容器所在ECS。

说明

- 单个VPC中内默认支持50个不同网段的容器部署，如须扩大需要申请扩大VPC路由表数目。
- 容器迁移到其他弹性云服务器后，需要在VPC路由表中添加新的路由信息。

步骤5 添加安全组规则。

为了能够通过traceroute命令和ping命令测试容器网络是否连通，为弹性云服务器的安全组添加如**表8-1**所示规则，开放ICMP和UDP规则。

具体操作请参见**添加安全组规则**。

表 8-1 安全组规则

方向	协议/应用	端口	源地址
入方向	ICMP	全部	0.0.0.0/0
入方向	UDP	全部	0.0.0.0/0

----结束

配置验证

分别在两台弹性云服务器上部署容器，通过ping来测试容器网络是否能连通。

以使用Docker部署容器为例，在ECS1上先创建一个网络连接my-net并指定容器网段为10.0.2.0/24，然后创建容器并指定使用my-net。

```
$ docker network create  --subnet 10.0.2.0/24 my-net
$ docker run -d --name nginx --net my-net -p 8080:80  nginx:alpine
```

在ECS2上同样创建网络连接和容器，容器网段为10.0.3.0/24。

```
$ docker network create  --subnet 10.0.3.0/24 my-net
$ docker run -d --name nginx --net my-net -p 8080:80  nginx:alpine
```

将弹性云服务器上iptables里面filter表的FORWARD链的默认策略设置为ACCEPT。

说明

此处设置是因为Docker为安全性将iptables里面filter表的FORWARD链默认策略设置成了drop，因此需要修改。

```
$ iptables -P FORWARD ACCEPT
```

进入10.0.2.2这个容器，尝试ping和traceroute 10.0.3.2，可以看到能够ping通，且traceroute路由路径为10.0.2.2 -> 10.0.2.1 -> 192.168.0.3 -> 10.0.3.2，与前面设置的路由转发规则一致。

```
[root@ecs1 ~]# docker exec -it nginx /bin/sh
/ # traceroute -d 10.0.3.2
traceroute to 10.0.3.2 (10.0.3.2), 30 hops max, 46 byte packets
 1  10.0.2.1 (10.0.2.1)  0.007 ms  0.004 ms  0.007 ms
 2  192.168.0.3 (192.168.0.3)  0.232 ms  0.165 ms  0.248 ms
 3  10.0.3.2 (10.0.3.2)  0.366 ms  0.308 ms  0.158 ms
/ # ping 10.0.3.2
PING 10.0.3.2 (10.0.3.2): 56 data bytes
64 bytes from 10.0.3.2: seq=0 ttl=62 time=0.570 ms
64 bytes from 10.0.3.2: seq=1 ttl=62 time=0.343 ms
```



```
64 bytes from 10.0.3.2: seq=2 ttl=62 time=0.304 ms
64 bytes from 10.0.3.2: seq=3 ttl=62 time=0.319 ms
```


9 使用虚拟 IP 和 Keepalived 搭建高可用 Web 集群

应用场景

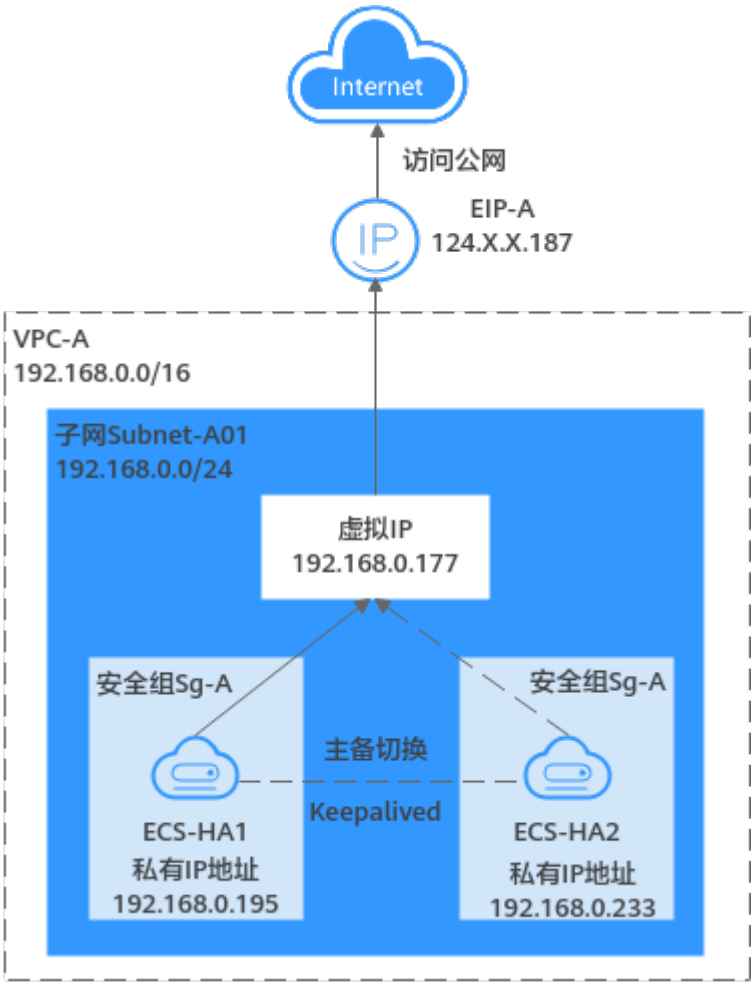
虚拟IP（Virtual IP Address）是从VPC子网网段中划分的一个内网IP地址，通常搭配高可用软件（比如Keepalived）使用，主要用来搭建高可用的主备集群。多个云服务器形成主备集群，当主云服务器发生故障无法对外提供服务时，系统动态将虚拟IP切换到备云服务器，通过备云服务器继续对外提供服务。本文档为您详细介绍使用虚拟IP和Keepalived搭建高可用Web集群的方法。

方案架构

本示例中，高可用Web集群架构如图9-1所示，将虚拟IP同时绑定至ECS-HA1和ECS-HA2，使用Keepalived搭建一个高可用集群。同时，为虚拟IP绑定EIP，该集群具备公网访问能力，可以面向公网提供Web访问服务。实现原理如下：

1. ECS-HA1作为主云服务器，通过与虚拟IP绑定的EIP对外提供服务，ECS-HA2作为备云服务器不承载实际业务。
2. 当ECS-HA1发生故障时，此时会自动启用ECS-HA2，ECS-HA2将会接管业务并对外提供服务，实现业务不中断的高可用需求。

图 9-1 使用虚拟 IP 和 Keepalived 搭建高可用 Web 集群



方案优势

基于虚拟IP和Keepalived能力，采用“一主一备”或“一主多备”的方法组合使用云服务器，这些云服务器对外呈现为一个虚拟IP。当主云服务器故障时，备云服务器可以转为主云服务器并继续对外提供服务，以此达到高可用性HA（High Availability）的目的，可以解决用户由于云服务器故障导致的对外服务中断问题。

约束与限制

使用虚拟IP搭建的高可用集群，所有服务器必须位于同一个虚拟私有云的子网内。

资源规划说明

本示例中，虚拟私有云VPC和子网、虚拟IP、弹性公网IP以及弹性云服务器ECS等资源只要位于同一个区域内即可，可用区可以任意选择，无需保持一致。

说明

以下资源规划详情仅为示例，您可以根据需要自行修改。

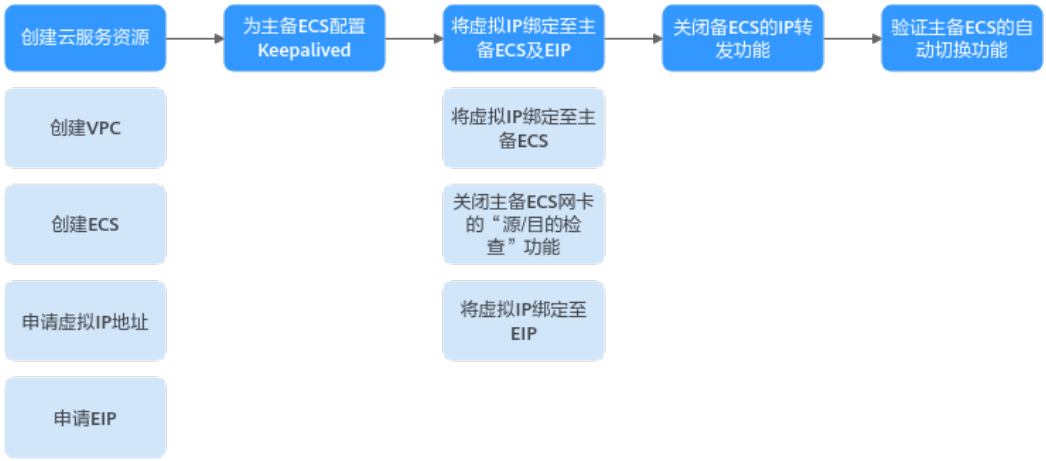
表 9-1 使用虚拟 IP 和 Keepalived 搭建高可用 Web 集群资源规划总体说明

资源类型	资源数量	说明
虚拟私有云VPC和子网	1	• VPC名称：请根据实际情况填写，本示例为VPC-A。 • IPv4网段：请根据实际情况填写，本示例为192.168.0.0/16。 • 子网名称：请根据实际情况填写，本示例为Subnet-A01。 • 子网IPv4网段：请根据实际情况填写，本示例为192.168.0.0/24。
弹性云服务器ECS	2	本示例中，需要两个ECS作为主备倒换，配置说明如下： • 名称：根据实际情况填写，本示例分别为ECS-HA1和ECS-HA2。 • 镜像：请根据实际情况选择，本示例为公共镜像（CentOS 7.8 64bit）。 • 系统盘：通用型SSD盘，40GB。 • 数据盘：本示例未选购数据盘，请您根据实际业务需求选购数据盘，并切实考虑两个ECS节点之间的业务数据一致性问题。 • 网络： – 虚拟私有云：选择您的虚拟私有云，本示例为VPC-A。 – 子网：选择子网，本示例为Subnet-A01。 • 安全组：请根据实际情况选择，本示例中ECS-HA1和ECS-HA2使用同一个安全组，安全组名称为Sg-A。 • 私有IP地址：ECS-HA1为192.168.0.195，ECS-HA2为192.168.0.233
虚拟IP	1	在子网Subnet-A01中申请虚拟IP地址： • 创建方式：根据实际情况填写，本示例为自动分配。 • 虚拟IP地址：本示例为192.168.0.177。 • 绑定实例：将虚拟IP绑定至ECS-HA1和ECS-HA2。 • 绑定弹性公网IP：将虚拟IP绑定至EIP-A。
弹性公网IP	1	• 计费模式：请根据情况选择计费模式，本示例为按需计费。 • EIP名称：请根据实际情况填写，本示例为EIP-A。 • EIP地址：EIP地址系统随机分配，本示例为124.X.X.187。

操作流程

使用虚拟IP和Keepalived搭建高可用Web集群，流程如图9-2所示。

图 9-2 使用虚拟 IP 和 Keepalived 搭建高可用 Web 集群



步骤一：创建云服务资源

1. 创建1个VPC和1个子网。

   具体方法请参见**创建虚拟私有云和子网**。

2. 创建2个ECS，分别作为主ECS和备ECS。

   具体方法请参见**购买方式概述**。

   本示例中，ECS的网络配置详情如下：

   - 网络：选择已创建的虚拟私有云和子网，VPC-A和Subnet-A01。
   - 安全组：新建一个安全组Sg-A，并添加入方向和出方向规则。您在创建安全组的时候，系统会自动添加部分规则，您需要根据实际情况进行检查修改。

     本示例中，ECS-HA1和ECS-HA2属于同一个安全组，您需要确保**表9-2**中的规则均已正确添加。

图 9-3 安全组入方向

图 9-4 安全组出方向

表 9-2 安全组 Sg-A 规则说明

方向	策略	类型	协议端口	源地址/目的地址	描述
入方向	允许	IPv4	TCP:22	源地址：0.0.0.0/0	放通安全组内ECS的SSH(22)端口，用于远程登录Linux ECS。
入方向	允许	IPv4	TCP:3389	源地址：0.0.0.0/0	放通安全组内ECS的RDP(3389)端口，用于远程登录Windows ECS。
入方向	允许	IPv4	TCP:80	源地址：0.0.0.0/0	放通安全组内ECS的HTTP(80)端口，用于外部通过HTTP协议访问ECS上部署的网站。
入方向	允许	IPv4	全部	源地址：当前安全组Sg-A	针对IPv4，用于安全组内ECS之间网络互通。
入方向	允许	IPv6	全部	源地址：当前安全组Sg-A	针对IPv6，用于安全组内ECS之间网络互通。
出方向	允许	IPv4	全部	目的地址：0.0.0.0/0	针对IPv4，用于安全组内ECS访问外部，允许流量从安全组内ECS流出。
出方向	允许	IPv6	全部	目的地址：::/0	针对IPv6，用于安全组内ECS访问外部，允许流量从安全组内ECS流出。

须知

本示例中，源地址设置为0.0.0.0/0表示允许所有外部IP远程登录云服务器，为了确保安全，建议您遵循最小原则，根据实际情况将源IP设置为特定的IP地址，比如，源地址设置为您的本地PC地址。

如果您的ECS位于不同的安全组，比如ECS-HA1属于Sg-A，ECS-HA2属于Sg-B，则除了在两个安全组中分别配置表9-2中的规则，您还需要添加表9-3中的规则，放通两个安全组之间的内网网络流量。

表 9-3 安全组 Sg-A 和 Sg-B 规则说明

安全组	方向	策略	类型	协议端口	源地址/目的地址	描述
Sg-A	入方向	允许	IPv4	全部	源地址：Sg-B	针对全部IPv4协议，允许来自Sg-B内实例的流量访问Sg-A内的实例。

安全组	方向	策略	类型	协议端口	源地址/目的地址	描述
Sg-B	入方向	允许	IPv4	全部	源地址：Sg-A	针对全部IPv4协议，允许来自Sg-A内实例的流量访问Sg-B内的实例。

- 弹性公网IP：选择“暂不购买”。

3. 在子网Subnet-A01内，申请虚拟IP地址。
   具体方法请参见**申请虚拟IP地址**。
4. 申请弹性公网IP。
   具体方法请参见**购买弹性公网IP**。

步骤二：为主备 ECS 配置 Keepalived

1. 执行以下操作，为ECS-HA1配置Keepalived。
   a. 将EIP绑定至ECS-HA1。
      具体方法请参见**绑定弹性公网IP**。
   b. 远程登录ECS-HA1。
      ECS有多种登录方法，具体请参见**登录弹性云服务器**。
   c. 执行以下命令，安装Nginx、Keepalived软件包及相关依赖包。
      yum install nginx keepalived -y
      回显类似如下信息，表示安装完成。

```
[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. 执行以下操作，修改Nginx配置文件，添加80端口相关配置。

i. 执行以下命令，打开“/etc/nginx/nginx.conf”文件。

vim /etc/nginx/nginx.conf

ii. 按**i**进入编辑模式。

iii. 将文件中原有的内容，全部替换成以下内容。

```
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. 按**ESC**退出，并输入**:wq!**保存配置。

e. 执行以下操作，修改index.html文件内容，用来验证网站的访问情况。

i. 执行以下命令，打开“/usr/share/nginx/html/index.html”文件。

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

ii. 按**i**进入编辑模式。

iii. 将文件中原有的内容，全部替换成以下内容。

```
Welcome to ECS-HA1
```

iv. 按**ESC**退出，并输入**:wq!**保存配置。

f. 执行以下命令，设置Nginx服务开机自启动，并启动Nginx服务。

systemctl enable nginx

systemctl start nginx.service

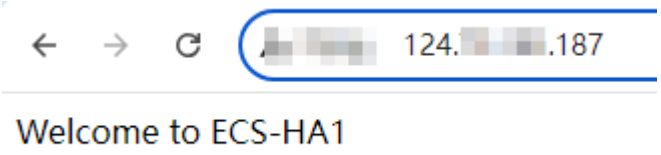
回显类似如下信息：

```
[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. 打开浏览器，并输入EIP地址（124.X.X.187），验证Nginx单节点的访问情况。

网页如下图所示，表示ECS-HA1的Nginx配置成功。

图 9-5 ECS-HA1 访问验证



h. 执行以下操作，修改Keepalived配置文件。

i. 执行以下命令，打开“/etc/keepalived/keepalived.conf”文件。

vim /etc/keepalived/keepalived.conf

ii. 按**i**进入编辑模式。

iii. 根据实际情况，替换配置文件中的IP参数，并将文件中原有的内容，全部替换成以下内容。

- mcast_src_ip和unicast_src_ip：替换为ECS-HA1的私有IP地址，本示例为192.168.0.195。
- virtual_ipaddress：替换为虚拟IP地址，本示例为192.168.0.177。

```
! 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. 按**ESC**退出，并输入**:wq!**保存配置。

i. 执行以下操作，配置Nginx监控脚本。

i. 执行以下命令，打开“/etc/keepalived/chk_nginx.sh”文件。

vim /etc/keepalived/chk_nginx.sh

ii. 按**i**进入编辑模式。

iii. 将文件中原有的内容，全部替换成以下内容。

```
#!/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. 按**ESC**退出，并输入**:wq!**保存配置。

j. 执行以下命令，为“chk_nginx.sh”文件添加执行权限。

chmod +x /etc/keepalived/chk_nginx.sh

k. 执行以下命令，设置Keepalived服务开机自启动，并启动Keepalived服务。

systemctl enable keepalived

systemctl start keepalived.service

l. 将EIP和ECS-HA1解绑定。

具体方法请参见**解绑弹性公网IP**。

2. 执行以下操作，为ECS-HA2配置Keepalived。

a. 将EIP绑定至ECS-HA2。

具体方法请参见**绑定弹性公网IP**。

b. 远程登录ECS-HA2。

ECS有多种登录方法，具体请参见**登录弹性云服务器**。

c. 执行以下命令，安装Nginx、Keepalived软件包及相关依赖包。

yum install nginx keepalived -y

回显类似如下信息，表示安装完成。

```
[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. 执行以下操作，修改Nginx配置文件，添加80端口相关配置。

i. 执行以下命令，打开“/etc/nginx/nginx.conf”文件。

vim /etc/nginx/nginx.conf

ii. 按**i**进入编辑模式。

iii. 将文件中原有的内容，全部替换成以下内容。

```
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. 按**ESC**退出，并输入**:wq!**保存配置。

e. 执行以下操作，修改index.html文件内容，用来验证网站的访问情况。

i. 执行以下命令，打开“/usr/share/nginx/html/index.html”文件。

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

ii. 按**i**进入编辑模式。

iii. 将文件中原有的内容，全部替换成以下内容。

```
Welcome to ECS-HA2
```

iv. 按**ESC**退出，并输入**:wq!**保存配置。

f. 执行以下命令，设置Nginx服务开机自启动，并启动Nginx服务。

systemctl enable nginx

systemctl start nginx.service

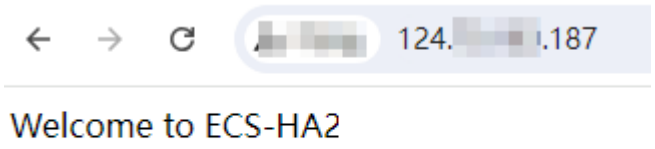
回显类似如下信息：

```
[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. 打开浏览器，并输入EIP地址（124.X.X.187），验证Nginx单节点的访问情况。

网页如下图所示，表示ECS-HA2的Nginx配置成功。

图 9-6 ECS-HA2 访问验证



h. 执行以下操作，修改Keepalived配置文件。

i. 执行以下命令，打开“/etc/keepalived/keepalived.conf”文件。

vim /etc/keepalived/keepalived.conf

ii. 按**i**进入编辑模式。

iii. 根据实际情况，替换配置文件中的IP参数，并将文件中原有的内容，全部替换成以下内容。

- mcast_src_ip和unicast_src_ip：替换为ECS-HA2的私有IP地址，本示例为192.168.0.233。
- virtual_ipaddress：替换为虚拟IP地址，本示例为192.168.0.177。

```
! 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. 按**ESC**退出，并输入**:wq!**保存配置。

i. 执行以下操作，配置Nginx监控脚本。

i. 执行以下命令，打开“/etc/keepalived/chk_nginx.sh”文件。

vim /etc/keepalived/chk_nginx.sh

ii. 按**i**进入编辑模式。

iii. 将文件中原有的内容，全部替换成以下内容。

```
#!/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. 按**ESC**退出，并输入**:wq!**保存配置。

j. 执行以下命令，为“chk_nginx.sh”文件添加执行权限。

chmod +x /etc/keepalived/chk_nginx.sh

k. 执行以下命令，设置Keepalived服务开机自启动，并启动Keepalived服务。

systemctl enable keepalived

systemctl start keepalived.service

l. 将EIP和ECS-HA2解绑定。

具体方法请参见**解绑弹性公网IP**。

步骤三：将虚拟 IP 绑定至主备 ECS 及 EIP

1. 将虚拟IP分别绑定至主备ECS上，本示例中需要绑定ECS-HA1和ECS-HA2。

   具体操作请参见**将虚拟IP地址绑定至实例或EIP**。

2. 关闭主备ECS网卡的“源/目的检查”功能。

   将虚拟IP绑定至ECS时，系统会自动关闭ECS网卡的“源/目的检查”功能，您需要参考以下操作检查关闭情况。如果未关闭，则请关闭该功能。

   a. 在ECS列表中，单击目标ECS的名称。

   进入ECS详情页。

   b. 选择“弹性网卡”页签，并单击 ⌄ 展开ECS的网卡详情区域，可以查看“源/目的检查”功能。

   如**图9-7**所示，表示“源/目的检查”功能已关闭。

图 9-7 关闭网卡的“源/目的检查”功能

3. 将虚拟IP绑定至EIP上，本示例中需要绑定EIP-A。
   具体操作请参见**将虚拟IP地址绑定至实例或EIP**。

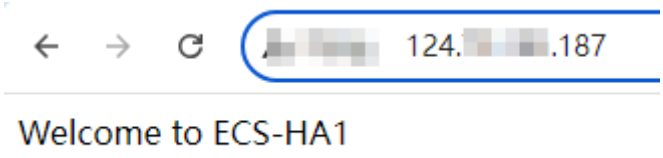
步骤四：关闭备 ECS 的 IP 转发功能

使用虚拟IP构建主备场景的高可用集群时，需要关闭备ECS的IP转发功能，当主备ECS切换后，则需要确保新的备ECS也关闭IP转发功能。

为了避免ECS主备切换后遗漏配置，建议您将主备ECS的IP转发功能全都关闭。

1. 打开浏览器，并输入EIP地址（124.X.X.187），通过网页确认主ECS。
   网页如下图所示，表示此时主ECS是ECS-HA1。

   图 9-8 主 ECS 验证

2. 远程登录备ECS，本示例是ECS-HA2。
   ECS有多种登录方法，具体请参见**登录弹性云服务器**。
3. 请根据ECS的操作系统，在**表9-4**中选择关闭IP转发功能的操作，本示例ECS为Linux操作系统。

表 9-4 关闭 IP 转发功能

操作系统	操作指导
Linux系统	1. 执行以下命令，切换root用户。 **su root** 2. 执行以下命令，查看IP转发功能是否已开启。 **cat /proc/sys/net/ipv4/ip_forward** 回显结果：1为开启，0为关闭，默认为0。 ● 回显为0，任务结束。 ● 回显为1，继续执行以下操作。 3. 以下提供两种方法修改配置文件，二选一即可。 方法一： a. 执行以下命令，打开“/etc/sysctl.conf”文件。 **vim /etc/sysctl.conf** b. 按**i**进入编辑模式。 c. 修改net.ipv4.ip_forward = 0。 d. 按**ESC**退出，并输入**:wq!**保存配置。 方法二： 执行sed命令，命令示例如下： **sed -i '/net.ipv4.ip_forward/s/1/0/g' /etc/sysctl.conf** 4. 执行以下命令，使修改生效。 **sysctl -p /etc/sysctl.conf**
Windows系统	1. 在搜索框中输入**cmd**，打开Windows系统的“命令提示符”窗口，执行以下命令。 **ipconfig/all** ● 回显结果中，“IP 路由已启用”为 “否”， 表示IP转发功能已关闭。 ● 回显结果中，“IP 路由已启用”为 “是”， 表示IP转发功能未关闭，继续执行以下操作。 2. 在搜索框中输入**regedit**，打开注册表编辑器。 3. 编辑**HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Tcpip\Parameters**下的**IPEnableRouter**值为0。 ● 指定值为 0：关闭 IP 转发。 ● 指定值为 1：启用 IP 转发。

步骤五：验证主备 ECS 的自动切换功能

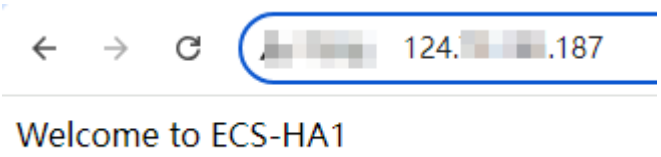
1. 执行以下操作，分别重启主备ECS。
   a. 远程登录ECS-HA1。
      ECS有多种登录方法，具体请参见**登录弹性云服务器**。
   b. 执行以下命令，重启ECS-HA1。
      reboot

c. 参考1.a~1.b，重启ECS-HA2。

2. 执行以下操作，验证主ECS的网页访问情况。

a. 打开浏览器，并输入EIP地址（124.X.X.187），验证主ECS的网页访问情况。
网页如下图所示，表示此时主ECS是ECS-HA1，且网页访问正常。

图 9-9 主 ECS 验证（ECS-HA1）



b. 远程登录ECS-HA1，并执行以下命令，查看虚拟IP是否已绑定到ECS-HA1的eth0网卡上。

ip addr show

回显类似如下信息，可以看到虚拟IP（192.168.0.177）已绑定至eth0网卡上，再次确认ECS-HA1为主ECS。

```
[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. 执行以下命令，停止主ECS的Keepalived服务，本示例中主ECS为ECS-HA1。

systemctl stop keepalived.service

3. 执行以下命令，验证主ECS是否切换成ECS-HA2。

a. 远程登录ECS-HA2，并执行以下命令，查看虚拟IP是否已绑定到ECS-HA2的eth0网卡上。

ip addr show

回显类似如下信息，可以看到虚拟IP（192.168.0.177）已绑定至eth0网卡上，此时确认ECS-HA2为主ECS。

```
[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
```

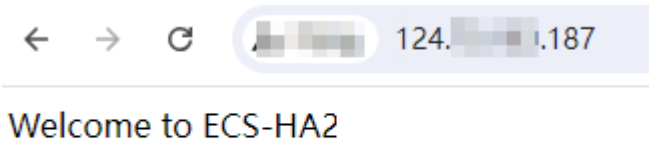


```
    valid_lft forever preferred_lft forever
inet6 fe80::f816:3eff:fefe:563f/64 scope link
    valid_lft forever preferred_lft forever
```

b. 打开浏览器，并输入EIP地址（124.X.X.187），验证ECS-HA2作为主ECS时的网页访问情况。

网页如下图所示，表示此时主ECS是ECS-HA2，且网页访问正常。

图 9-10 主 ECS 验证（ECS-HA2）



Welcome to ECS-HA2

10 为多网卡 ECS 配置策略路由

10.1 方案概述

背景知识

当云服务器拥有多张网卡时，主网卡默认可以和外部正常通信，扩展网卡无法和外部正常通信，此时需要在云服务器内部为这些网卡配置策略路由，才可以确保多张网卡均可以和外部正常通信。

操作场景

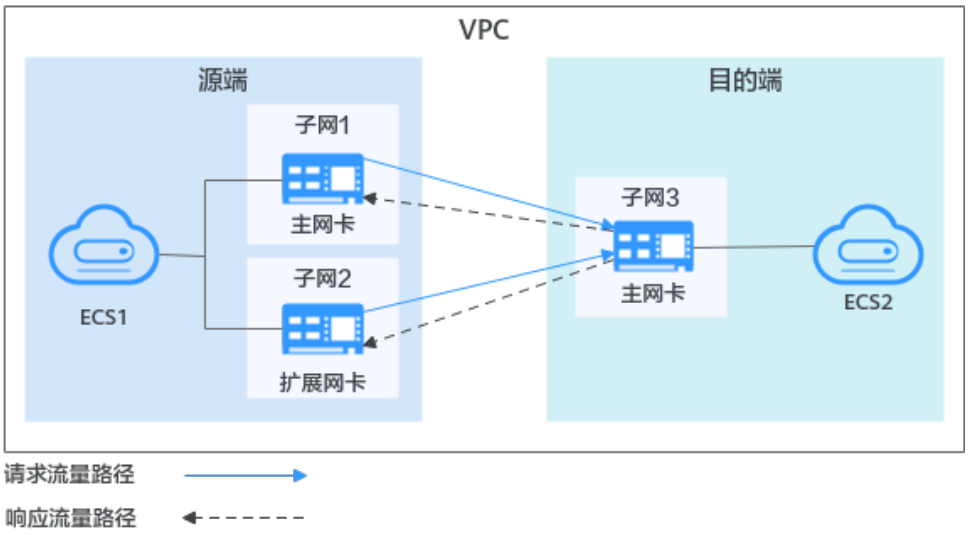
本文档以配置双网卡云服务器的策略路由为例，组网如图10-1所示，具体说明如下：

- 源端云服务器主网卡和扩展网卡位于同一个VPC内的不同子网。
- 源端云服务器和目的端云服务器位于同一个VPC内的不同子网，因此网络互通，即配置策略路由前，源端云服务器的主网卡可以和目的端云服务器正常通信。
- 为源端云服务器双网卡配置策略路由后，主网卡和扩展网卡都可以作为独立网卡和目的端云服务器正常通信。

须知

您可以根据实际情况选择目的端地址，请在配置双网卡策略路由前，确保源端云服务器主网卡和目的端已正常通信。

图 10-1 双网卡云服务器组网示意图



操作指引

本文提供Linux和Windows云服务器的操作指导，具体请参见**表10-1**。

表 10-1 操作指引说明

操作系统类型	IP类型	操作步骤
Linux	IPv4	本文以CentOS 8.0 64bit操作系统为例： **为多网卡Linux云服务器配置策略路由（IPv4/IPv6）**
	IPv6	
Windows	IPv4	本文以Windows 2012 64bit操作系统为例： **为多网卡Windows云服务器配置策略路由（IPv4/IPv6）**
	IPv6	

10.2 收集云服务器网络信息

操作场景

为多网卡云服务器配置策略路由之前，您需要收集云服务器的网络信息，请根据云服务器操作系统及IP类型参考对应的指导，具体说明如下：

- 对于Linux IPv4场景，您需要收集的信息如**表10-2**所示。

表 10-2 Linux IPv4 场景信息说明

类型	主网卡	扩展网卡	获取方法
源端	• 网卡地址：10.0.0.115 • 子网网段：10.0.0.0/24 • 子网网关：10.0.0.1	• 网卡地址：10.0.1.183 • 子网网段：10.0.1.0/24 • 子网网关：10.0.1.1	• **获取云服务器网卡地址** • **获取子网网段和网关地址**
目的端	网卡地址：10.0.2.12	不涉及	

- 对于Linux IPv6场景，您需要收集的信息如**表10-3**所示。

表 10-3 Linux IPv6 场景信息说明

类型	主网卡	扩展网卡	获取方法
源端	• IPv4网卡地址：10.0.0.102 • IPv6网卡地址：2407:c080:1200:1dd8:859c:e5d5:8b3d:a2d9 • 子网IPv6网段：2407:c080:1200:1dd8::/64 • 子网IPv6网关：2407:c080:1200:1dd8::1	• IPv4网卡地址：10.0.1.191 • IPv6网卡地址：2407:c080:1200:1a9c:7cc0:63b5:8e65:4dd8 • 子网IPv6网段：2407:c080:1200:1a9c::/64 • 子网IPv6网关：2407:c080:1200:1a9c::1	• **获取云服务器网卡地址** • **获取子网网段和网关地址**
目的端	• IPv4网卡地址：10.0.2.3 • IPv6网卡地址：2407:c080:1200:1dd9:16a7:fe7a:8f71:7044	不涉及	

- 对于Windows IPv4场景，您需要收集的信息如**表10-4**所示。

表 10-4 Windows IPv4 场景信息说明

类型	主网卡	扩展网卡	获取方法
源端	• 网卡地址：10.0.0.59 • 子网网关：10.0.0.1	• 网卡地址：10.0.1.104 • 子网网关：10.0.1.1	• **获取云服务器网卡地址** • **获取子网网段和网关地址**

类型	主网卡	扩展网卡	获取方法
目的端	网卡地址：10.0.2.12	不涉及	

- 对于Windows IPv6场景，您需要收集的信息如**表10-5**所示。

表 10-5 Windows IPv6 场景信息说明

类型	主网卡	扩展网卡	获取方法
源端	网卡地址：2407:c080:802:aba:6788:fb94:d71f:8deb	网卡地址：2407:c080:802:be6:71c8:42e0:d44e:eeb4	**获取云服务器网卡地址**
目的端	网卡地址：2407:c080:802:be7:c2e6:d99c:b685:c6c8	不涉及	

获取云服务器网卡地址

1. 登录管理控制台。
2. 在管理控制台左上角单击 ，选择区域和项目。
3. 在服务列表，选择“计算 > 弹性云服务器”。
4. 在弹性云服务器列表中，选择目标弹性云服务器，并单击名称对应的超链接。
   进入弹性云服务器“基本信息”页签。
5. 在网卡区域，查看云服务器主网卡和扩展网卡对应IP地址。
   支持查看IPv4及IPv6地址。

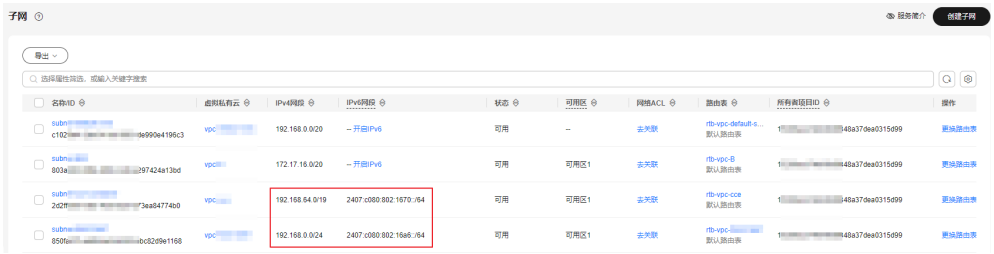
图 10-2 查看云服务器网卡地址

获取子网网段和网关地址

1. 登录管理控制台。

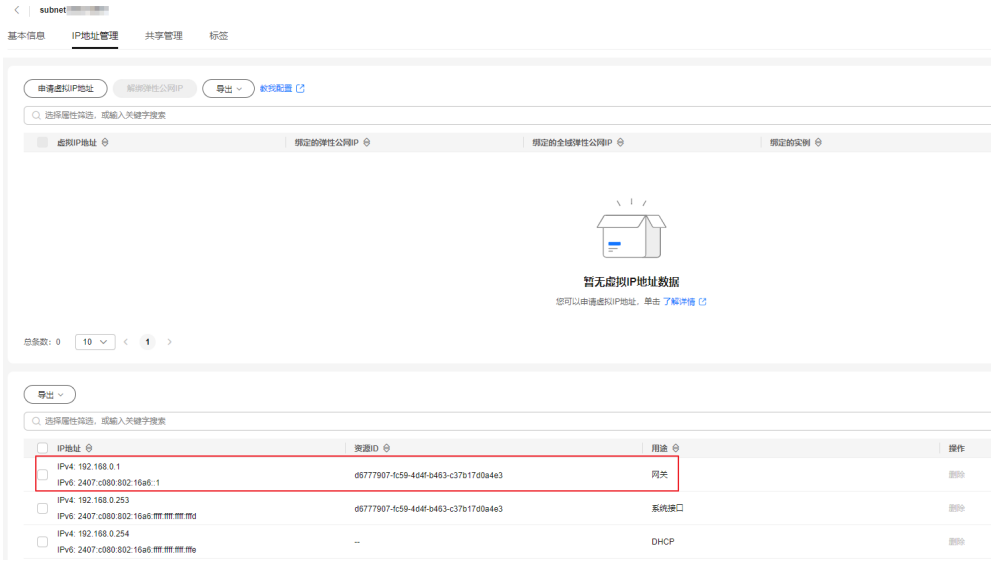
2. 在管理控制台左上角单击 ，选择区域和项目。
3. 在服务列表，选择“计算 > 弹性云服务器”。
4. 在弹性云服务器列表中，选择目标弹性云服务器，并单击名称对应的超链接。
   进入弹性云服务器“基本信息”页签。
5. 在云服务器信息区域，单击虚拟私有云对应的超链接。
   进入“虚拟私有云”页面。
6. 在虚拟私有云列表中，单击“子网个数”所在列的数字。
   进入“子网”页面。
7. 在子网列表中，查看目标子网对应的网段。
   支持查看IPv4和IPv6地址。

图 10-3 查看子网网段



8. 在子网列表中，单击子网名称对应的超链接。
   进入子网的“基本信息”页面。
9. 选择“IP地址管理”页签，查看目标子网对应的网关地址。
   支持查看IPv4和IPv6地址。

图 10-4 查看子网对应的网关地址



10.3 为多网卡 Linux 云服务器配置策略路由（IPv4/IPv6）

操作场景

本文档以CentOS 8.0 64bit为例，指导用户为双网卡Linux云服务器配置策略路由。

- IPv4：**操作步骤 (Linux IPv4)**
- IPv6：**操作步骤 (Linux IPv6)**

关于云服务器双网卡的背景知识及组网说明，请参见**方案概述**。

操作步骤 (Linux IPv4)

1. 收集配置策略路由需要的云服务器网卡地址等信息。
   具体操作请参见**收集云服务器网络信息**。
2. 登录弹性云服务器。
   弹性云服务器有多种登录方法，具体请参见**登录弹性云服务器**。
3. 执行以下命令，检查源端云服务器主网卡和目的端云服务器通信情况。
   ping -I *源端云服务器主网卡地址 目的端云服务器地址*
   命令示例：
   ping -I 10.0.0.115 10.0.2.12
   回显类似如下信息，表示可以正常通信。

```
[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 ---
```

   说明

   配置多网卡策略路由前，请务必确保源端主网卡和目的端通信正常。

4. 执行以下命令，查看云服务器网卡名称。
   ifconfig
   回显类似如下信息，通过网卡地址查找对应的网卡名称，本示例中：
   - 10.0.0.115为主网卡地址，对应的名称为eth0。
   - 10.0.1.183为扩展网卡地址，对应的名称为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. 执行以下步骤，为云服务器配置临时路由。

须知

临时路由配置完后立即生效，当云服务器重启后临时路由会丢失，请执行完5配置完临时路由后，继续执行6配置永久路由，避免云服务器重启后网络中断。

a. 依次执行以下命令，添加主网卡和扩展网卡的策略路由。

- 主网卡
  ip route add default via *子网网关* **dev** *网卡名称* **table** *路由表名称*
  ip route add *子网网段* **dev** *网卡名称* **table** *路由表名称*
  ip rule add from *网卡地址* **table** *路由表名称*
- 扩展网卡
  ip route add default via *子网网关* **dev** *网卡名称* **table** *路由表名称*
  ip route add *子网网段* **dev** *网卡名称* **table** *路由表名称*
  ip rule add from *网卡地址* **table** *路由表名称*

参数说明如下：

- 网卡名称：填写4中所查名称。
- 路由表名称：自定义路由表名称，此处请使用数字命名路由表。
- 其他网络信息：填写1中收集的地址。

命令示例：

- 主网卡
  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
- 扩展网卡
  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

说明

如果云服务器有多张网卡，请依次为所有网卡添加策略路由。

b. 依次执行以下命令，确认策略路由是否添加成功。
ip rule
ip route show table *主网卡路由表名称*
ip route show table *扩展网卡路由表名称*
其中，路由表名称为5.a中自定义的名称。

命令示例：

ip rule

ip route show table 10

ip route show table 20

回显类似如下信息，表示策略路由添加成功。

```
[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. 执行以下命令，验证源端云服务器和目的端云服务器是否可以正常通信。

ping -I *源端云服务器主网卡地址 目的端云服务器地址*

ping -I *源端云服务器扩展网卡地址 目的端云服务器地址*

命令示例：

ping -I 10.0.0.115 10.0.2.12

ping -I 10.0.1.183 10.0.2.12

回显类似如下信息，两个网卡均可以和目的端正常通信，表示策略路由配置成功。

```
[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. 执行以下步骤，为云服务器配置永久路由。

a. 执行以下命令，打开“/etc/rc.local”文件。

vi /etc/rc.local

b. 按**i**进入编辑模式。

c. 在文件末尾添加以下配置。

```
# wait for nics up
sleep 5
# 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
```

```
# 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
```

其中，参数说明如下：

- wait for nics up：文件启动时间，建议和本示例中的配置保持一致。
- Add v4 routes for eth0：主网卡的策略路由，和5.a配置保持一致。
- Add v4 routes for eth1：扩展网卡的策略路由，和5.a配置保持一致。
- Add v4 routes for cloud-init：配置cloud-init地址，请和本示例中的配置保持一致，不要修改。

d. 按**ESC**退出，并输入**:wq!**保存配置。

e. 执行以下命令，为“/etc/rc.local”文件添加执行权限。

chmod +x /etc/rc.local

说明

如果您的操作系统为Redhat、EulerOS，执行完6.e后，还需要执行以下命令，权限才会添加成功。

chmod +x /etc/rc.d/rc.local

f. 执行以下命令，重启云服务器。

reboot

须知

“/etc/rc.local”文件添加中添加的策略路由，需要重启云服务器后才会生效，此处请确保不影响业务再重启云服务器操作。

g. 参考5.b~5.c，检查策略路由添加情况，并验证源端和目的通信是否正常。

操作步骤 (Linux IPv6)

1. 收集配置策略路由需要的云服务器网卡地址等信息。

   具体操作请参见**收集云服务器网络信息**。

2. 登录弹性云服务器。

   弹性云服务器有多种登录方法，具体请参见**登录弹性云服务器**。

3. 执行以下步骤，确保云服务器已开启IPv6协议栈，并且正常获取到IPv6地址。

须知

对于源端和目的端的IPv6云服务器，均需要执行该操作，确保云服务器已获取到IPv6地址，否则云服务器无法通过IPv6地址进行通信。

本章节云服务使用的操作系统为CentOS 8.0 64bit公共镜像，以下针对该操作系统举例，更多操作系统配置指导，请参见**动态获取IPv6地址**的“Linux操作系统（手动配置启用IPv6）” 小节。

a. 执行以下命令，检查云服务器是否可以获取到IPv6地址。

ip addr

回显类似如下信息，eth0和eth1为云服务器的网卡，只有一行inet6地址，为fe80开头，表示该云服务器已开启IPv6协议栈，但是未获取到IPv6地址，需要继续执行**3.b**~**3.g**，获取IPv6地址。

```
[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. 执行以下命令，查看云服务器网卡名称。

ifconfig

回显类似如下信息，通过网卡地址查找对应的网卡名称，本示例中：

- 10.0.0.102为主网卡地址，对应的名称为eth0。
- 10.0.1.191为扩展网卡地址，对应的名称为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. 执行以下步骤，编辑主网卡的ifcfg文件。

i. 执行以下命令，打开主网卡的ifcfg文件。

vi /etc/sysconfig/network-scripts/ifcfg-*主网卡名称*

其中，主网卡名称为**3.b**中查询到的名称。

命令示例：

vi /etc/sysconfig/network-scripts/ifcfg-eth0

ii. 按**i**进入编辑模式。

iii. 在文件末尾添加以下配置。

```
IPV6INIT="yes"
DHCPV6C="yes"
```

iv. 按**ESC**退出，并输入**:wq!**保存配置。

d. 执行以下步骤，编辑扩展网卡的ifcfg文件。

i. 执行以下命令，打开扩展网卡的ifcfg文件。

vi /etc/sysconfig/network-scripts/ifcfg-*扩展网卡名称*

其中，扩展网卡名称为**3.b**中查询到的名称。

命令示例：

vi /etc/sysconfig/network-scripts/ifcfg-eth1

ii. 按**i**进入编辑模式。

iii. 在文件末尾添加以下配置。

```
IPV6INIT="yes"
DHCPV6C="yes"
```

iv. 按**ESC**退出，并输入**:wq!**保存配置。

e. 执行以下步骤，编辑“/etc/sysconfig/network”文件。

i. 执行以下命令，打开“/etc/sysconfig/network”文件。

vi /etc/sysconfig/network

ii. 按**i**进入编辑模式。

iii. 在文件末尾添加以下配置。

```
NETWORKING_IPV6="yes"
```

iv. 按**ESC**退出，并输入**:wq!**保存配置。

f. 执行以下命令，重启网络服务使配置生效。

systemctl restart NetworkManager

g. 执行以下命令，检查云服务器是否可以获取到IPv6地址。

ip addr

回显类似如下信息，eth0和eth1网卡有两行inet6地址，新增一行2407开头的地址，表示配置成功。

```
[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
```

4. 执行以下命令，检查源端云服务器主网卡和目的端云服务器通信情况。

ping6 -I *源端云服务器主网卡地址 目的端云服务器地址*

命令示例：

ping6 -I 2407:c080:1200:1dd8:859c:e5d5:8b3d:a2d9 2407:c080:1200:1dd9:16a7:fe7a:8f71:7044

回显类似如下信息，表示可以正常通信。

```
[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
```

说明

配置多网卡策略路由前，请务必确保源端主网卡和目的端通信正常。

5. 执行以下步骤，为云服务器配置临时路由。

须知

临时路由配置完后立即生效，当云服务器重启后临时路由会丢失，请执行5配置完临时路由后，继续执行6配置永久路由，避免云服务器重启后网络中断。

a. 依次执行以下命令，添加主网卡和扩展网卡的策略路由。

- 主网卡

  ip -6 route add default via *子网网关* **dev** *网卡名称* **table** *路由表名称*

  ip -6 route add *子网网段* **dev** *网卡名称* **table** *路由表名称*

  ip -6 rule add from *网卡地址* **table** *路由表名称*

- 扩展网卡

  ip -6 route add default via *子网网关* **dev** *网卡名称* **table** *路由表名称*

  ip -6 route add *子网网段* **dev** *网卡名称* **table** *路由表名称*

  ip -6 rule add from *网卡地址* **table** *路由表名称*

参数说明如下：

- 网卡名称：填写3.b中所查名称。
- 路由表名称：自定义路由表名称，此处请使用数字命名路由表。
- 其他网络信息：填写1中收集的地址。

命令示例：

- 主网卡

  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

- 扩展网卡

  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

说明

如果云服务器有多张网卡，请依次为所有网卡添加策略路由。

b. 依次执行以下命令，确认策略路由是否添加成功。

ip -6 rule

ip -6 route show table *主网卡路由表名称*

ip -6 route show table *扩展网卡路由表名称*

其中，路由表名称为**5.a**中自定义的名称。

命令示例：

ip -6 rule

ip -6 route show table 10

ip -6 route show table 20

回显类似如下信息，表示策略路由添加成功。

```
[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. 执行以下命令，验证源端云服务器和目的端云服务器是否可以正常通信。

ping -6 -I *源端云服务器主网卡地址 目的端云服务器地址*

ping -6 -I *源端云服务器扩展网卡地址 目的端云服务器地址*

命令示例：

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

回显类似如下信息，两个网卡均可以和目的端正常通信，表示策略路由配置成功。

```
[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. 执行以下步骤，为云服务器配置永久路由。
   a. 执行以下命令，打开“/etc/rc.local”文件。

      vi /etc/rc.local

   b. 按**i**进入编辑模式。
   c. 在文件末尾添加以下配置。

```
# wait for nics up
sleep 5
# 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
# 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
```

      其中，参数说明如下：

      - wait for nics up：文件启动时间，建议和本示例中的配置保持一致。
      - Add v6 routes for eth0：主网卡的策略路由，和**5.a**配置保持一致。
      - Add v6 routes for eth1：扩展网卡的策略路由，和**5.a**配置保持一致。

   d. 按**ESC**退出，并输入**:wq!**保存配置。
   e. 执行以下命令，为“/etc/rc.local”文件添加执行权限。

      chmod +x /etc/rc.local

      说明

      如果您的操作系统为Redhat、EulerOS，执行完**6.e**后，还需要执行以下命令，权限才会添加成功。

      chmod +x /etc/rc.d/rc.local

   f. 执行以下命令，重启云服务器。

      reboot

      须知

      “/etc/rc.local”文件添加中添加的策略路由，需要重启云服务器后才会生效，此处请确保不影响业务再重启云服务器操作。

   g. 参考**5.b**~**5.c**，检查策略路由添加情况，并验证源端和目的通信是否正常。

10.4 为多网卡 Windows 云服务器配置策略路由（IPv4/IPv6）

操作场景

本文档以Windows 2012 64bit为例，指导用户为双网卡Windows云服务器配置策略路由。

- IPv4：**操作步骤 (Windows IPv4)**
- IPv6：**操作步骤 (Windows IPv6)**

关于云服务器双网卡的背景知识及组网说明，请参见**方案概述**。

操作步骤 (Windows IPv4)

1. 收集配置策略路由需要的云服务器网卡地址等信息。
   具体操作请参见**收集云服务器网络信息**。
2. 登录弹性云服务器。
   弹性云服务器有多种登录方法，具体请参见**登录弹性云服务器**。
3. 执行以下命令，检查源端云服务器主网卡和目的端云服务器通信情况。
   ping -S *源端云服务器主网卡地址 目的端云服务器地址*
   命令示例：
   ping -S 10.0.0.59 10.0.2.12
   回显类似如下信息，表示可正常通信。

   ```
   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
   ```

   说明

   配置多网卡策略路由前，请务必确保源端主网卡和目的端通信正常。
4. 执行以下命令，添加扩展网卡的策略路由。
   route add -p 0.0.0.0 mask 0.0.0.0 *扩展网卡子网网关* **metric** *路由优先级*
   参数说明如下：
   - **0.0.0.0/0**：默认路由，请不要修改。
   - 扩展网卡子网网关：填写1中收集的地址。
   - 路由优先级：扩展网卡优先级必须小于主网卡，数字越大优先级越低，此处配置为261。

   命令示例：
   route add -p 0.0.0.0 mask 0.0.0.0 10.0.1.1 metric 261

说明

- 主网卡策略路由系统已有，不需要添加。
- 如果云服务器有多张扩展网卡，请依次为所有扩展网卡添加策略路由。

5. 执行以下命令，确认策略路由是否添加成功。

   route print

   回显类似如下信息，表示策略路由添加成功。该路由为永久路由，重启后不会丢失。

```
C:\Users\Administrator>route print
===========================================================================
Interface List
 19...fa 16 3e fc 7b 76 ......Red Hat VirtIO Ethernet Adapter #3
 14...fa 16 3e 5d 3e b6 ......Red Hat VirtIO Ethernet Adapter
  1...........................Software Loopback Interface 1
 16...00 00 00 00 00 00 00 e0 Microsoft ISATAP Adapter #2
===========================================================================

IPv4 Route Table
===========================================================================
Active Routes:
Network Destination        Netmask          Gateway       Interface  Metric
          0.0.0.0          0.0.0.0         10.0.1.1       10.0.1.104     266
          0.0.0.0          0.0.0.0         10.0.0.1        10.0.0.59       5
         10.0.0.0    255.255.255.0         On-link         10.0.0.59     261
        10.0.0.59  255.255.255.255         On-link         10.0.0.59     261
       10.0.0.255  255.255.255.255         On-link         10.0.0.59     261
         10.0.1.0    255.255.255.0         On-link       10.0.1.104     261
       10.0.1.104  255.255.255.255         On-link       10.0.1.104     261
       10.0.1.255  255.255.255.255         On-link       10.0.1.104     261
        127.0.0.0        255.0.0.0         On-link        127.0.0.1     306
        127.0.0.1  255.255.255.255         On-link        127.0.0.1     306
  127.255.255.255  255.255.255.255         On-link        127.0.0.1     306
  169.254.169.254  255.255.255.255        10.0.0.254        10.0.0.59       6
        224.0.0.0        240.0.0.0         On-link        127.0.0.1     306
        224.0.0.0        240.0.0.0         On-link         10.0.0.59     261
        224.0.0.0        240.0.0.0         On-link       10.0.1.104     261
  255.255.255.255  255.255.255.255         On-link        127.0.0.1     306
  255.255.255.255  255.255.255.255         On-link         10.0.0.59     261
  255.255.255.255  255.255.255.255         On-link       10.0.1.104     261
===========================================================================
Persistent Routes:
  Network Address          Netmask  Gateway Address  Metric
          0.0.0.0          0.0.0.0         10.0.1.1     261
===========================================================================

IPv6 Route Table
===========================================================================
Active Routes:
 If Metric Network Destination      Gateway
  1    306 ::1/128                  On-link
 14    261 fe80::/64                On-link
 19    261 fe80::/64                On-link
 19    261 fe80::197b:3504:e05:5a4d/128
                                    On-link
 14    261 fe80::e115:8e6a:5dcc:6715/128
                                    On-link
  1    306 ff00::/8                 On-link
 14    261 ff00::/8                 On-link
 19    261 ff00::/8                 On-link
===========================================================================
Persistent Routes:
  None
```

6. 执行以下命令，验证源端云服务器和目的端云服务器是否可以正常通信。

   ping -S *源端云服务器主网卡地址 目的端云服务器地址*

   ping -S *源端云服务器扩展网卡地址 目的端云服务器地址*

   命令示例：

   ping -S 10.0.0.59 10.0.2.12

   ping -S 10.0.1.104 10.0.2.12

   回显类似如下信息，两个网卡均可以和目的端正常通信，表示策略路由配置成功。


```
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

Ping statistics for 10.0.2.12:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\Users\Administrator>ping -S 10.0.1.104 10.0.2.12

Pinging 10.0.2.12 from 10.0.1.104 with 32 bytes of data:
Reply from 10.0.2.12: bytes=32 time=4ms 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

Ping statistics for 10.0.2.12:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 4ms, Average = 1ms
```

操作步骤 (Windows IPv6)

1. 收集配置策略路由需要的云服务器网卡地址等信息。

   具体操作请参见**收集云服务器网络信息**。

2. 登录弹性云服务器。

   弹性云服务器有多种登录方法，具体请参见**登录弹性云服务器**。

3. 执行以下命令，确保云服务器已开启IPv6协议栈，并且正常获取到IPv6地址。

   ipconfig

   回显类似如下信息，每个网卡可以查看到IPv6地址，为2407开头的地址，表示可以自动获取到IPv6地址，不用进行配置。

```
C:\Users\Administrator>ipconfig

Windows IP Configuration


Ethernet adapter Ethernet 4:

   Connection-specific DNS Suffix  . : openstacklocal
   IPv6 Address. . . . . . . . . . . : 2407:c080:802:be6:ec23:ec4:c886:cc1
   Link-local IPv6 Address . . . . . : fe80::8836:a693:1b03:a17d%19
   IPv4 Address. . . . . . . . . . . : 192.168.1.12
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Default Gateway . . . . . . . . . : fe80::f816:3eff:fe3e:1e1e%19

Ethernet adapter Ethernet 2:

   Connection-specific DNS Suffix  . : openstacklocal
   IPv6 Address. . . . . . . . . . . : 2407:c080:802:aba:8999:5e61:e19:cf7e
   Link-local IPv6 Address . . . . . : fe80::180d:f3b5:27ac:2acb%14
   IPv4 Address. . . . . . . . . . . : 192.168.0.57
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Default Gateway . . . . . . . . . : fe80::f816:3eff:fede:c837%14
                                       192.168.0.1

Tunnel adapter isatap.openstacklocal:

   Media State . . . . . . . . . . . : Media disconnected
   Connection-specific DNS Suffix  . : openstacklocal

C:\Users\Administrator>_
```


须知

对于源端和目的端的IPv6云服务器，均需要执行该操作，确保云服务器已获取到IPv6地址，否则云服务器无法通过IPv6地址进行通信。

本章节云服务器使用的操作系统为Windows 2012 64bit公共镜像，无需额外配置，云服务器可自动获取获取到IPv6地址。如果您的云服务器无法自动获取到IPv6地址，请参见**动态获取IPv6地址**的“Windows 2012操作系统”和“Windows 2008操作系统”小节进行配置。

4. 执行以下命令，验证源端云服务器和目的端云服务器是否可以正常通信。

   ping -6 -S *源端云服务器主网卡地址 目的端云服务器地址*

   ping -6 -S *源端云服务器扩展网卡地址 目的端云服务器地址*

   命令示例：

   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

   回显类似如下信息，两个网卡均可以和目的端正常通信，表示策略路由配置成功。

```
C:\Users\Administrator>ping -6 -S 2407:c080:802:aba:8999:5e61:e19:cf7e 2407:c080
:802:be7:c2e6:d99c:b685:c6c8

Pinging 2407:c080:802:be7:c2e6:d99c:b685:c6c8 from 2407:c080:802:aba:8999:5e61:e
19:cf7e with 32 bytes of data:
Reply from 2407:c080:802:be7:c2e6:d99c:b685:c6c8: time<1ms
Reply from 2407:c080:802:be7:c2e6:d99c:b685:c6c8: time<1ms
Reply from 2407:c080:802:be7:c2e6:d99c:b685:c6c8: time<1ms
Reply from 2407:c080:802:be7:c2e6:d99c:b685:c6c8: time<1ms

Ping statistics for 2407:c080:802:be7:c2e6:d99c:b685:c6c8:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\Users\Administrator>ping -6 -S 2407:c080:802:be6:ec23:ec4:c886:cc1 2407:c080:
802:be7:c2e6:d99c:b685:c6c8

Pinging 2407:c080:802:be7:c2e6:d99c:b685:c6c8 from 2407:c080:802:be6:ec23:ec4:c8
86:cc1 with 32 bytes of data:
Reply from 2407:c080:802:be7:c2e6:d99c:b685:c6c8: time=3ms
Reply from 2407:c080:802:be7:c2e6:d99c:b685:c6c8: time<1ms
Reply from 2407:c080:802:be7:c2e6:d99c:b685:c6c8: time<1ms
Reply from 2407:c080:802:be7:c2e6:d99c:b685:c6c8: time<1ms

Ping statistics for 2407:c080:802:be7:c2e6:d99c:b685:c6c8:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 3ms, Average = 0ms
```

须知

本章节示例中，云服务器使用Windows 2012 64bit公共镜像，对于IPv6场景，不需要在云服务器内配置策略路由，双网卡可正常通信。

11 连通不同 VPC 网络的对等连接配置示例

11.1 对等连接配置示例概述

不同的VPC之间内网隔离，您可以使用对等连接将两个VPC连通起来，对等连接两端的VPC使用私有IP地址进行通信。对等连接只可以连通同区域VPC，如果您需要连通不同区域的VPC，请参见**云连接**。

本文档为您提供了不同组网的VPC对等连接配置示例，包括IPv4和IPv6，请您根据VPC资源情况选择相应的指导。

表 11-1 对等连接资源情况说明

资源情况	对等连接配置示例
• VPC：VPC网段不重叠 • 子网：子网网段不重叠	您可以创建整个VPC网段之间的对等连接，即对等连接两端连通的是整个VPC。 组网示例，请参见**指向整个VPC网段的对等连接配置**。
• VPC：VPC网段重叠 • 子网：部分子网网段重叠	VPC网段重叠时，您无法创建整个VPC网段之间的对等连接，此时建议您创建VPC子网之间、VPC内ECS之间的对等连接，具体说明如下： • VPC子网之间的对等连接，即对等连接连通的是不同VPC的子网。需要确保对等连接两端的子网网段不能重叠。 组网示例，请参见**指向VPC子网的对等连接配置**。 • VPC内ECS之间的对等连接，即对等连接连通的是不同VPC内的ECS。需要确保对等连接两端ECS的私有IP地址不能相同。 组网示例，请参见**指向VPC内ECS的对等连接配置**。

资源情况	对等连接配置示例
• VPC：VPC网段重叠 • 子网：全部子网网段重叠	您无论是创建指向整个VPC网段，还是子网网段或者ECS的对等连接，均是无效的，此场景下不支持使用VPC对等连接。 详细说明，请参见**无效的VPC对等连接配置**。

11.2 连通整个 VPC 网络的对等连接配置示例

您可以参考以下示例，配置连通整个VPC网络的对等连接，在VPC路由表中添加的路由目的地址为对端VPC网段，此时通过对等连接可以连通整个VPC内的所有资源，示例场景如**表11-2**所示。

表 11-2 指向整个 VPC 网段的对等连接场景说明

组网示例	推荐场景	IP类型	配置示例
相互对等的两个VPC	当您需要两个VPC之间彼此资源互访时，可以参考本示例规划组网。 比如，人力资源部门使用VPC-A，财务部门使用VPC-B，需要这两个VPC之间资源无限制互访。	IPv4	**配置相互对等的两个VPC (IPv4)**
		IPv6	**配置相互对等的两个VPC (IPv6)**
相互对等的多个VPC	当您需要多个VPC之间彼此资源互访时，可以参考本示例规划组网。 比如，人力资源部门使用VPC-A，财务部门使用VPC-B，市场部门使用VPC-C，需要多个VPC之间资源无限制互访	IPv4	**配置相互对等的多个VPC (IPv4)**
		IPv4	**基于对等连接的传递性配置相互对等的多个VPC(IPv4)**
		IPv6	**配置相互对等的多个VPC (IPv6)**
一个中心VPC与两个VPC对等	当您需要一个中心VPC和其他两个VPC之间资源互访时，要求其他两个VPC可以访问中心VPC的所有资源，但彼此之间隔离时，可以参考本示例规划组网。 比如，您的中心VPC-A上部署有公共服务（例如数据库），VPC-B和VPC-C均需要访问该数据库，但是VPC-B和VPC-C之间无需资源互访。	IPv4	**配置一个中心VPC与两个VPC对等 (IPv4)**
		IPv6	**配置一个中心VPC与两个VPC对等 (IPv6)**
一个中心VPC的主网段和扩展网段与两个VPC对等	本示例与上述示例类似，只是中心VPC存在主网段和扩展网段。	IPv4	**配置一个中心VPC的主网段和扩展网段与两个VPC对等 (IPv4)**

组网示例	推荐场景	IP类型	配置示例
一个中心VPC与多个VPC对等	当您需要一个中心VPC和其他多个VPC之间资源互访时，要求其他多个VPC可以访问中心VPC的所有资源，但彼此之间隔离时，可以参考本示例规划组网。 比如，您的中心VPC-A上部署有公共服务（例如数据库），VPC-B、VPC-C、VPC-D、VPC-E、VPC-F以及VPC-G均需要访问该数据库，但是这些VPC之间无需资源互访。	IPv4	配置一个中心VPC与多个VPC对等 (IPv4)
		IPv6	配置一个中心VPC与多个VPC对等 (IPv6)

约束与限制

配置指向整个VPC网段的对等连接时，相互对等的VPC网段不能重叠，否则会导致对等连接不生效，详细示例请参见**VPC网段重叠可能导致对等连接不生效**。

即使您的VPC对等连接仅用于IPv6通信，VPC的IPv4网段也不能重叠。本章节所有示例中，对等连接两端VPC的IPv4网段均不重叠。

配置相互对等的两个 VPC（IPv4）

本示例中，在VPC-A和VPC-B之间创建对等连接Peering-AB，VPC-A和VPC-B的网段不能重叠。

- 资源规划详情，请参见**表11-3**。
- 对等连接关系，请参见**表11-4**。

图 11-1 相互对等的两个 VPC(IPv4)

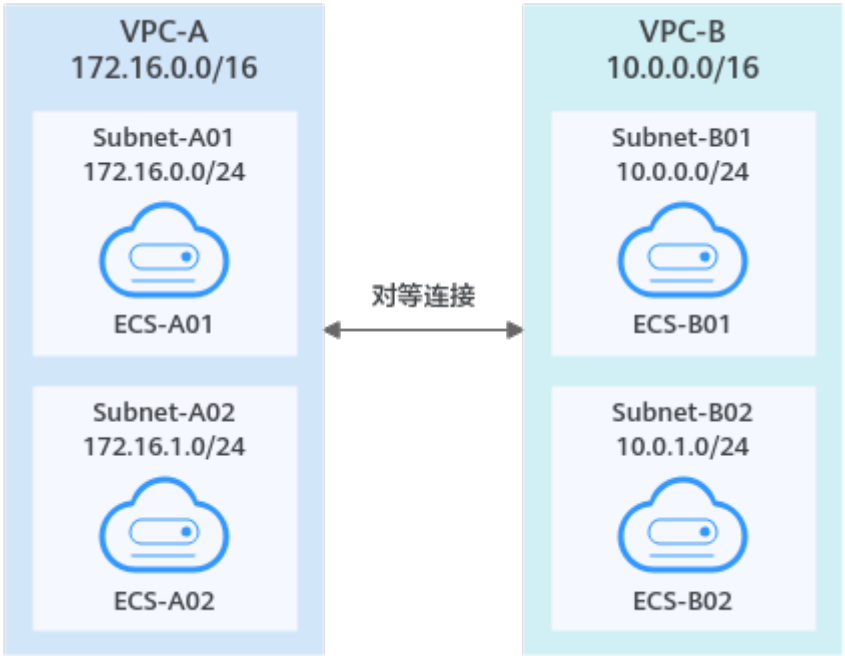


表 11-3 资源规划详情-相互对等的两个 VPC(IPv4)

VPC名称	VPC网段	子网名称	子网网段	子网关联VPC路由表	ECS名称	安全组	私有IP地址
VPC-A	172.16.0.0/16	Subnet-A01	172.16.0.0/24	rtb-VPC-A	ECS-A01	sg-web：通用Web服务器	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

表 11-4 对等连接关系说明-相互对等的两个 VPC(IPv4)

VPC对等关系	对等连接名称	本端VPC	对端VPC
VPC-A和VPC-B对等	Peering-AB	VPC-A	VPC-B

对等连接创建完成后，您需要在本端和对端VPC路由表中，添加以下路由：

表 11-5 VPC 路由表配置说明-相互对等的两个 VPC(IPv4)

路由表	目的地址	下一跳	路由类型	路由说明
rtb-VPC-A	172.16.0.0/24	Local	系统路由	Local路由是系统自动添加的，用于VPC内部通信。
	172.16.1.0/24	Local	系统路由	
	10.0.0.0/16 (VPC-B)	Peering-AB	自定义	在VPC-A的路由表中，添加目的地址为VPC-B网段，下一跳指向Peering-AB的路由。
rtb-VPC-B	10.0.0.0/24	Local	系统路由	Local路由是系统自动添加的，用于VPC内部通信。
	10.0.1.0/24	Local	系统路由	
	172.16.0.0/16 (VPC-A)	Peering-AB	自定义	在VPC-B的路由表中，添加目的地址为VPC-A网段，下一跳指向Peering-AB的路由。

说明

路由目的地址配置为对方VPC网段，可以访问对方VPC内的所有资源，VPC网段不能重叠。

配置相互对等的两个 VPC（IPv6）

本示例中，在VPC-A和VPC-B之间创建对等连接Peering-AB，VPC-A和VPC-B内的子网都具有IPv6网段，并且IPv4网段不能重叠。

- 资源规划详情，请参见表11-6。
- 对等连接关系，请参见表11-7。

图 11-2 相互对等的两个 VPC(IPv6)

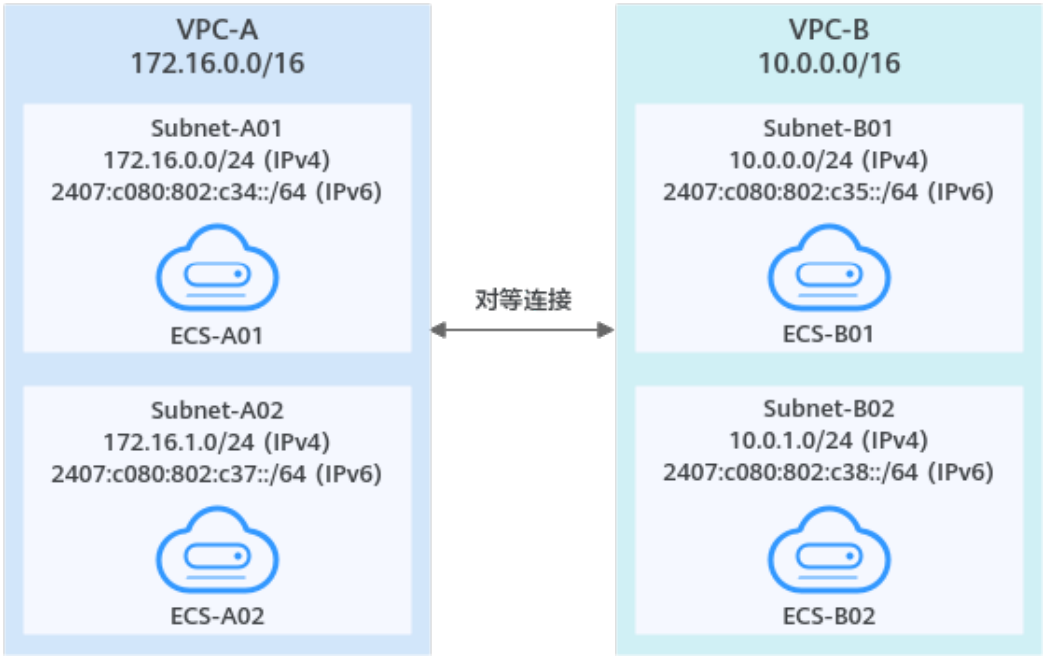


表 11-6 资源规划详情-相互对等的两个 VPC(IPv6)

VPC名称	VPC网段	子网名称	子网网段	子网关联VPC路由表	ECS名称	安全组	私有IP地址
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：通用Web服务器	• IPv4: 172.16.0.111 • IPv6: 2407:c080:802:c34:a925:f12e:cfa0:8edb

VPC名称	VPC网段	子网名称	子网网段	子网关联VPC路由表	ECS名称	安全组	私有IP地址
		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

表 11-7 对等连接关系说明-相互对等的两个 VPC(IPv6)

VPC对等关系	对等连接名称	本端VPC	对端VPC
VPC-A和VPC-B对等	Peering-AB	VPC-A	VPC-B

对等连接创建完成后，您需要在本端和对端VPC路由表中，添加以下路由：

表 11-8 VPC 路由表配置说明-相互对等的两个 VPC(IPv6)

路由表	目的地址	下一跳	路由类型	路由说明
rtb-VPC-A	172.16.0.0/24	Local	系统路由	Local路由是系统自动添加的，用于VPC内部通信。
	2407:c080:802:c34::/64	Local	系统路由	
	172.16.1.0/24	Local	系统路由	
	2407:c080:802:c37::/64	Local	系统路由	
	10.0.0.0/16 (VPC-B)	Peering-AB	自定义	在VPC-A的路由表中，添加目的地址为VPC-B网段，下一跳指向Peering-AB的路由，用于IPv4通信。
	2407:c080:802:c35::/64 (Subnet-B01)	Peering-AB	自定义	在VPC-A的路由表中，添加目的地址为子网Subnet-B01和Subnet-B02的IPv6网段，下一跳指向Peering-AB的路由，用于IPv6通信。
	2407:c080:802:c38::/64 (Subnet-B02)	Peering-AB	自定义	
rtb-VPC-B	10.0.0.0/24	Local	系统路由	Local路由是系统自动添加的，用于VPC内部通信。
	2407:c080:802:c35::/64	Local	系统路由	
	10.0.1.0/24	Local	系统路由	
	2407:c080:802:c38::/64	Local	系统路由	
	172.16.0.0/16 (VPC-A)	Peering-AB	自定义	在VPC-B的路由表中，添加目的地址为VPC-A网段，下一跳指向Peering-AB的路由，用于IPv4通信。
	2407:c080:802:c34::/64 (Subnet-A01)	Peering-AB	自定义	在VPC-B的路由表中，添加目的地址为子网Subnet-A01和Subnet-A02的IPv6网段，下一跳指向Peering-AB的路由，用于IPv6通信。
	2407:c080:802:c37::/64 (Subnet-A02)	Peering-AB	自定义	

说明

当前支持通过管理控制台获取VPC子网的IPv6地址，为了实现整个VPC资源互访，此处您需要逐次添加VPC内所有子网的IPv6网段。

配置相互对等的多个 VPC（IPv4）

本示例中，为了实现多个VPC通信，您需要在每个VPC之间两两建立对等连接，并且VPC-A、VPC-B和VPC-C的网段不能重叠。

- 资源规划详情，请参见**表11-9**。
- 对等连接关系，请参见**表11-10**。

图 11-3 相互对等的多个 VPC(IPv4)

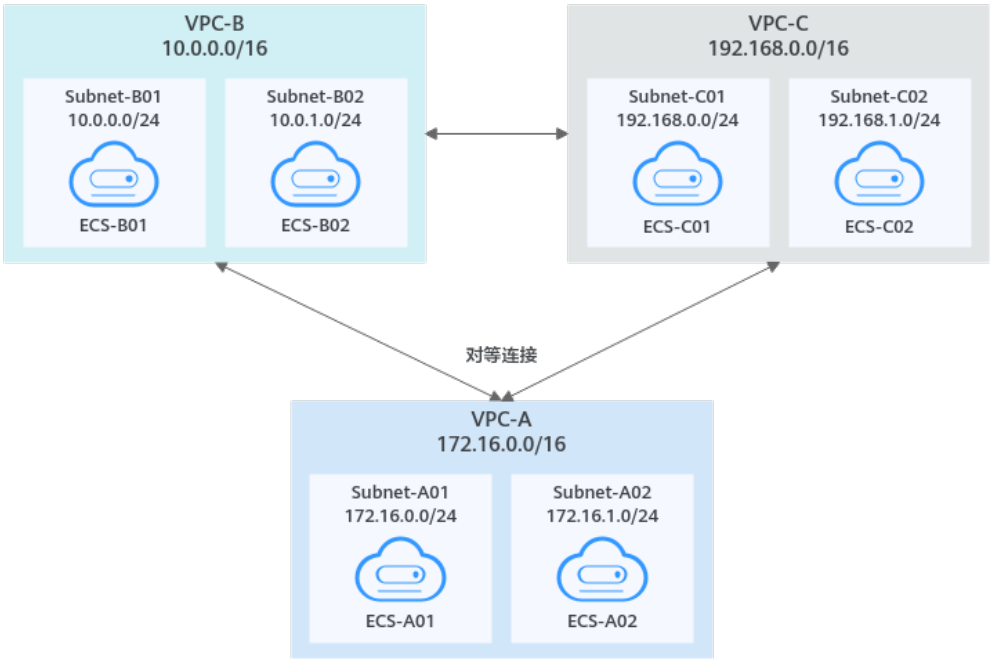


表 11-9 资源规划详情-相互对等的多个 VPC(IPv4)

VPC名称	VPC网段	子网名称	子网网段	子网关联VPC路由表	ECS名称	安全组	私有IP地址
VPC-A	172.16.0.0/16	Subnet-A01	172.16.0.0/24	rtb-VPC-A	ECS-A01	sg-web：通用Web服务器	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名称	VPC网段	子网名称	子网网段	子网关联VPC路由表	ECS名称	安全组	私有IP地址
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

表 11-10 对等连接关系说明-相互对等的多个 VPC(IPv4)

VPC对等关系	对等连接名称	本端VPC	对端VPC
VPC-A和VPC-B对等	Peering-AB	VPC-A	VPC-B
VPC-A和VPC-C对等	Peering-AC	VPC-A	VPC-C
VPC-B和VPC-C对等	Peering-BC	VPC-B	VPC-C

对等连接创建完成后，您需要在本端和对端VPC路由表中，添加以下路由：

表 11-11 VPC 路由表配置说明-相互对等的多个 VPC(IPv4)

路由表	目的地址	下一跳	路由类型	路由说明
rtb-VPC-A	172.16.0.0/24	Local	系统路由	Local路由是系统自动添加的，用于VPC内部通信。
	172.16.1.0/24	Local	系统路由	
	10.0.0.0/16 (VPC-B)	Peering-AB	自定义	在VPC-A的路由表中，添加目的地址为VPC-B网段，下一跳指向Peering-AB的路由。
	192.168.0.0/16 (VPC-C)	Peering-AC	自定义	在VPC-A的路由表中，添加目的地址为VPC-C网段，下一跳指向Peering-AC的路由。
rtb-VPC-B	10.0.0.0/24	Local	系统路由	Local路由是系统自动添加的，用于VPC内部通信。
	10.0.1.0/24	Local	系统路由	

路由表	目的地址	下一跳	路由类型	路由说明
	172.16.0.0/16 (VPC-A)	Peering-AB	自定义	在VPC-B的路由表中，添加目的地址为VPC-A网段，下一跳指向Peering-AB的路由。
	192.168.0.0/16 (VPC-C)	Peering-BC	自定义	在VPC-B的路由表中，添加目的地址为VPC-C网段，下一跳指向Peering-BC的路由。
rtb-VPC-C	192.168.0.0/24	Local	系统路由	Local路由是系统自动添加的，用于VPC内部通信。
	192.168.1.0/24	Local	系统路由	
	172.16.0.0/16 (VPC-A)	Peering-AC	自定义	在VPC-C的路由表中，添加目的地址为VPC-A网段，下一跳指向Peering-AC的路由。
	10.0.0.0/16 (VPC-B)	Peering-BC	自定义	在VPC-C的路由表中，添加目的地址为VPC-B网段，下一跳指向Peering-BC的路由。

说明

路由目的地址配置为对方VPC网段，可以访问对方VPC内的所有资源，VPC网段不能重叠。

基于对等连接的传递性配置相互对等的多个 VPC(IPv4)

对等连接具有传递性，如图11-4所示，在VPC-A和VPC-B、VPC-A和VPC-C之间创建对等连接，如果还需要实现VPC-B和VP-C之间的通信，您可以通过以下两种方案实现：

- 建立VPC-B和VPC-C之间的对等连接，具体配置请参见**配置相互对等的多个VPC (IPv4)**。
- 通过路由配置，可以基于VPC-A实现VPC-B和VPC-C之间的流量转发，具体请参见**表11-14**

图 11-4 对等连接具有传递性

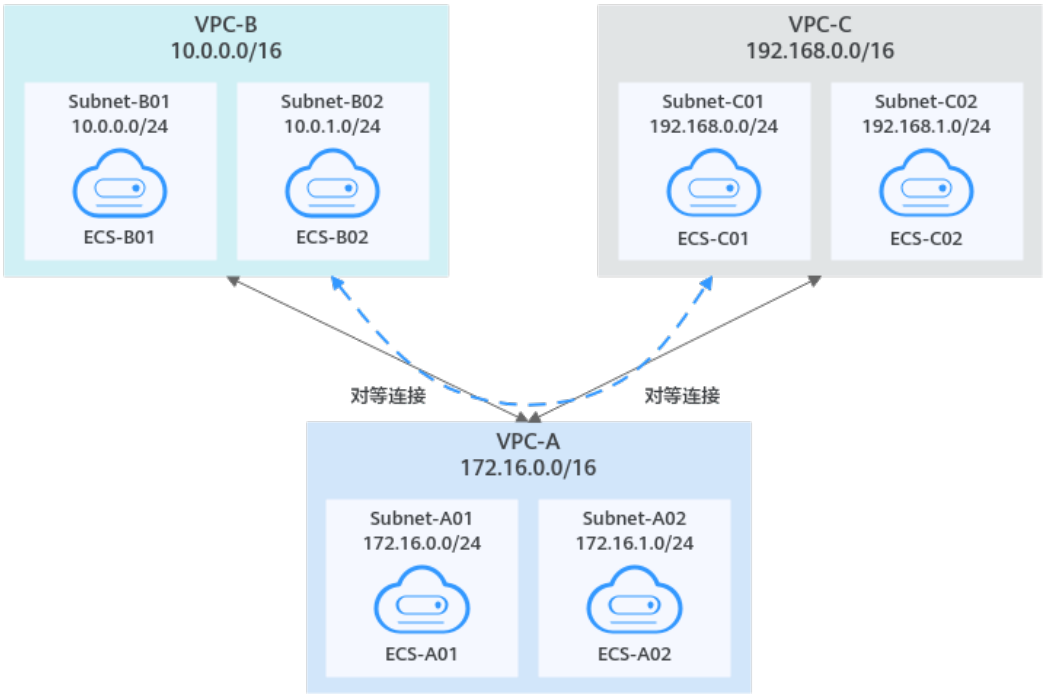


表 11-12 资源规划详情-基于对等连接的传递性配置相互对等的多个 VPC(IPv4)

VPC名称	VPC网段	子网名称	子网网段	子网关联VPC路由表	ECS名称	安全组	私有IP地址
VPC-A	172.16.0.0/16	Subnet-A01	172.16.0.0/24	rtb-VPC-A	ECS-A01	sg-web：通用Web服务器	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

表 11-13 对等连接关系说明-基于对等连接的传递性配置相互对等的多个 VPC(IPv4)

VPC对等关系	对等连接名称	本端VPC	对端VPC
VPC-A和VPC-B对等	Peering-AB	VPC-A	VPC-B

VPC对等关系	对等连接名称	本端VPC	对端VPC
VPC-A和VPC-C对等	Peering-AC	VPC-A	VPC-C

对等连接创建完成后，您需要在本端和对端VPC路由表中，添加以下路由：

表 11-14 VPC 路由表配置说明-基于对等连接的传递性配置相互对等的多个 VPC(IPv4)

路由表	目的地址	下一跳	路由类型	路由说明
rtb-VPC-A	172.16.0.0/24	Local	系统路由	Local路由是系统自动添加的，用于VPC内部通信。
	172.16.1.0/24	Local	系统路由	
	10.0.0.0/16 (VPC-B)	Peering-AB	自定义	在VPC-A的路由表中，添加目的地址为VPC-B网段，下一跳指向Peering-AB的路由。
	192.168.0.0/16 (VPC-C)	Peering-AC	自定义	在VPC-A的路由表中，添加目的地址为VPC-C网段，下一跳指向Peering-AC的路由。
rtb-VPC-B	10.0.0.0/24	Local	系统路由	Local路由是系统自动添加的，用于VPC内部通信。
	10.0.1.0/24	Local	系统路由	
	172.16.0.0/16 (VPC-A)	Peering-AB	自定义	在VPC-B的路由表中，添加目的地址为VPC-A网段，下一跳指向Peering-AB的路由。
	192.168.0.0/16 (VPC-C)	Peering-AB	自定义	在VPC-B的路由表中，添加目的地址为VPC-C网段，下一跳指向Peering-AB的路由。
rtb-VPC-C	192.168.0.0/24	Local	系统路由	Local路由是系统自动添加的，用于VPC内部通信。
	192.168.1.0/24	Local	系统路由	
	172.16.0.0/16 (VPC-A)	Peering-AC	自定义	在VPC-C的路由表中，添加目的地址为VPC-A网段，下一跳指向Peering-AC的路由。
	10.0.0.0/16 (VPC-B)	Peering-AC	自定义	在VPC-C的路由表中，添加目的地址为VPC-B网段，下一跳指向Peering-AC的路由。

说明

路由目的地址配置为对方VPC网段，可以访问对方VPC内的所有资源，VPC网段不能重叠。

配置相互对等的多个 VPC（IPv6）

本示例中，为了实现多个VPC通信，您需要在每个VPC之间两两建立对等连接，VPC-A、VPC-B和VPC-C内的子网都具有IPv6网段，并且VPC-A、VPC-B和VPC-C的IPv4网段不能重叠。

- 资源规划详情，请参见**表11-15**。
- 对等连接关系，请参见**表11-16**。

图 11-5 相互对等的多个 VPC(IPv6)

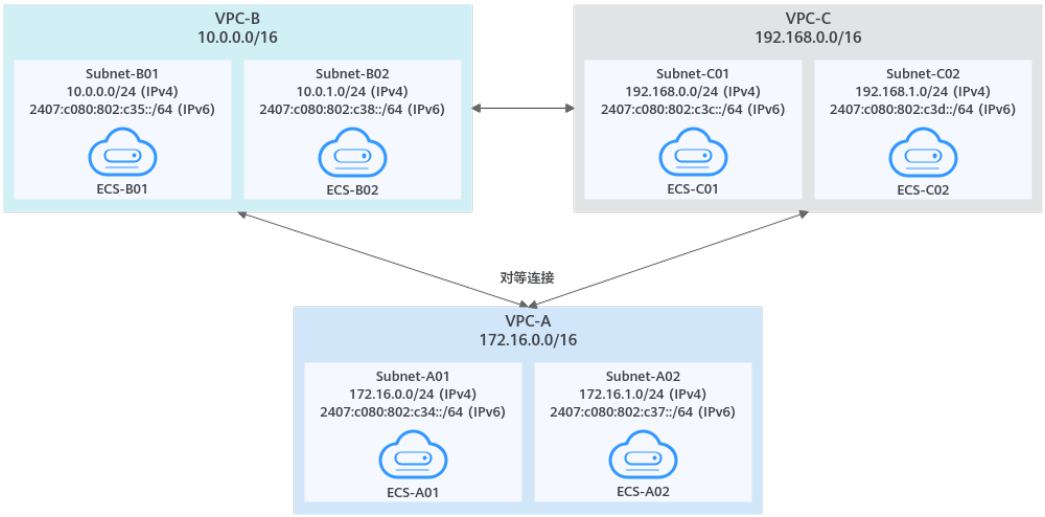


表 11-15 资源规划详情-相互对等的多个 VPC(IPv6)

VPC名称	VPC网段	子网名称	子网网段	子网关联VPC路由表	ECS名称	安全组	私有IP地址
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：通用Web服务器	• IPv4: 172.16.0.111 • IPv6: 2407:c080:802:c34:a925:f12e:cfa0:8edb

VPC名称	VPC网段	子网名称	子网网段	子网关联VPC路由表	ECS名称	安全组	私有IP地址
		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-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名称	VPC网段	子网名称	子网网段	子网关联VPC路由表	ECS名称	安全组	私有IP地址
		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

表 11-16 对等连接关系说明-相互对等的多个 VPC(IPv6)

VPC对等关系	对等连接名称	本端VPC	对端VPC
VPC-A和VPC-B对等	Peering-AB	VPC-A	VPC-B
VPC-A和VPC-C对等	Peering-AC	VPC-A	VPC-C
VPC-B和VPC-C对等	Peering-BC	VPC-B	VPC-C

对等连接创建完成后，您需要在本端和对端VPC路由表中，添加以下路由：

表 11-17 VPC 路由表配置说明-相互对等的多个 VPC(IPv6)

路由表	目的地址	下一跳	路由类型	路由说明
rtb-VPC-A	172.16.0.0/24	Local	系统路由	Local路由是系统自动添加的，用于VPC内部通信。
	2407:c080:802:c34::/64	Local	系统路由	
	172.16.1.0/24	Local	系统路由	
	2407:c080:802:c37::/64	Local	系统路由	

路由表	目的地址	下一跳	路由类型	路由说明
	10.0.0.0/16 (VPC-B)	Peering-AB	自定义	在VPC-A的路由表中，添加目的地址为VPC-B网段，下一跳指向Peering-AB的路由，用于IPv4通信。
	2407:c080:802:c35::/64 (Subnet-B01)	Peering-AB	自定义	在VPC-A的路由表中，添加目的地址为子网Subnet-B01和Subnet-B02的IPv6网段，下一跳指向Peering-AB的路由，用于IPv6通信。
	2407:c080:802:c38::/64 (Subnet-B02)	Peering-AB	自定义	
	192.168.0.0/16 (VPC-C)	Peering-AC	自定义	在VPC-A的路由表中，添加目的地址为VPC-C网段，下一跳指向Peering-AC的路由，用于IPv4通信。
	2407:c080:802:c3c::/64 (Subnet-C01)	Peering-AC	自定义	在VPC-A的路由表中，添加目的地址为子网Subnet-C01和Subnet-C02的IPv6网段，下一跳指向Peering-AC的路由，用于IPv6通信。
	2407:c080:802:c3d::/64 (Subnet-C02)	Peering-AC	自定义	
rtb-VPC-B	10.0.0.0/24	Local	系统路由	Local路由是系统自动添加的，用于VPC内部通信。
	2407:c080:802:c35::/64	Local	系统路由	
	10.0.1.0/24	Local	系统路由	
	2407:c080:802:c38::/64	Local	系统路由	
	172.16.0.0/16 (VPC-A)	Peering-AB	自定义	在VPC-B的路由表中，添加目的地址为VPC-A网段，下一跳指向Peering-AB的路由，用于IPv4通信。
	2407:c080:802:c34::/64 (Subnet-A01)	Peering-AB	自定义	在VPC-B的路由表中，添加目的地址为子网Subnet-A01和Subnet-A02的IPv6网段，下一跳指向Peering-AB的路由，用于IPv6通信。
	2407:c080:802:c37::/64 (Subnet-A02)	Peering-AB	自定义	
	192.168.0.0/16 (VPC-C)	Peering-BC	自定义	在VPC-B的路由表中，添加目的地址为VPC-C网段，下一跳指向Peering-BC的路由，用于IPv4通信。

路由表	目的地址	下一跳	路由类型	路由说明
	2407:c080:802:c3c::/64 (Subnet-C01)	Peering-BC	自定义	在VPC-B的路由表中，添加目的地址为子网Subnet-C01和Subnet-C02网段的IPv6，下一跳指向Peering-BC的路由，用于IPv6通信。
	2407:c080:802:c3d::/64 (Subnet-C02)	Peering-BC	自定义	
rtb-VPC-C	192.168.0.0/24	Local	系统路由	Local路由是系统自动添加的，用于VPC内部通信。
	2407:c080:802:c3c::/64	Local	系统路由	
	192.168.1.0/24	Local	系统路由	
	2407:c080:802:c3d::/64	Local	系统路由	
	172.16.0.0/16 (VPC-A)	Peering-AC	自定义	在VPC-C的路由表中，添加目的地址为VPC-A网段，下一跳指向Peering-AC的路由，用于IPv4通信。
	2407:c080:802:c34::/64 (Subnet-A01)	Peering-AC	自定义	在VPC-C的路由表中，添加目的地址为子网Subnet-A01和Subnet-A02的IPv6网段，下一跳指向Peering-AC的路由，用于IPv6通信。
	2407:c080:802:c37::/64 (Subnet-A02)	Peering-AC	自定义	
	10.0.0.0/16 (VPC-B)	Peering-BC	自定义	在VPC-C的路由表中，添加目的地址为VPC-B网段，下一跳指向Peering-BC的路由，用于IPv4通信。
	2407:c080:802:c35::/64 (Subnet-B01)	Peering-BC	自定义	在VPC-C的路由表中，添加目的地址为子网Subnet-B01和Subnet-B02的IPv6网段，下一跳指向Peering-BC的路由，用于IPv6通信。
	2407:c080:802:c38::/64 (Subnet-B02)	Peering-BC	自定义	

说明

当前支持通过管理控制台获取VPC子网的IPv6地址，为了实现整个VPC资源互访，此处您需要逐次添加VPC内所有子网的IPv6网段。

配置一个中心 VPC 与两个 VPC 对等（IPv4）

本示例中，在VPC-A和VPC-B之间创建对等连接Peering-AB，在VPC-A和VPC-C之间创建对等连接Peering-AC，并且VPC-A、VPC-B和VPC-C的网段不能重叠。

- 资源规划详情，请参见**表11-18**。
- 对等连接关系，请参见**表11-19**。

图 11-6 一个中心 VPC 与两个 VPC 对等(IPv4)

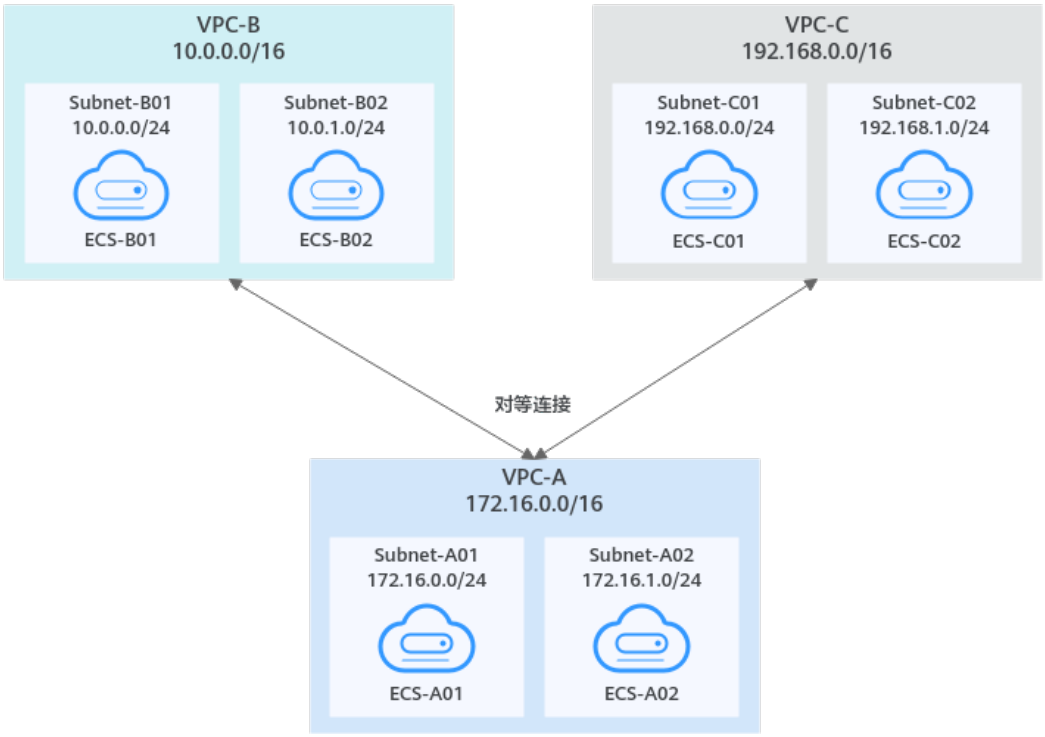


表 11-18 资源规划详情-一个中心 VPC 与两个 VPC 对等(IPv4)

VPC名称	VPC网段	子网名称	子网网段	子网关联VPC路由表	ECS名称	安全组	私有IP地址
VPC-A	172.16.0.0/16	Subnet-A01	172.16.0.0/24	rtb-VPC-A	ECS-A01	sg-web：通用Web服务器	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

VPC名称	VPC网段	子网名称	子网网段	子网关联VPC路由表	ECS名称	安全组	私有IP地址
		Subnet-C02	192.168.1.0/24	rtb-VPC-C	ECS-C02		192.168.1.200

表 11-19 对等连接关系说明-一个中心 VPC 与两个 VPC 对等(IPv4)

VPC对等关系	对等连接名称	本端VPC	对端VPC
VPC-A和VPC-B对等	Peering-AB	VPC-A	VPC-B
VPC-A和VPC-C对等	Peering-AC	VPC-A	VPC-C

对等连接创建完成后，您需要在本端和对端VPC路由表中，添加以下路由：

表 11-20 VPC 路由表配置说明-一个中心 VPC 与两个 VPC 对等(IPv4)

路由表	目的地址	下一跳	路由类型	路由说明
rtb-VPC-A	172.16.0.0/24	Local	系统路由	Local路由是系统自动添加的，用于VPC内部通信。
	172.16.1.0/24	Local	系统路由	
	10.0.0.0/16 (VPC-B)	Peering-AB	自定义	在VPC-A的路由表中，添加目的地址为VPC-B网段，下一跳指向Peering-AB的路由。
	192.168.0.0/16 (VPC-C)	Peering-AC	自定义	在VPC-A的路由表中，添加目的地址为VPC-C网段，下一跳指向Peering-AC的路由。
rtb-VPC-B	10.0.0.0/24	Local	系统路由	Local路由是系统自动添加的，用于VPC内部通信。
	10.0.1.0/24	Local	系统路由	
	172.16.0.0/16 (VPC-A)	Peering-AB	自定义	在VPC-B的路由表中，添加目的地址为VPC-A网段，下一跳指向Peering-AB的路由。
rtb-VPC-C	192.168.0.0/24	Local	系统路由	Local路由是系统自动添加的，用于VPC内部通信。

路由表	目的地址	下一跳	路由类型	路由说明
	192.168.1.0/24	Local	系统路由	
	172.16.0.0/16 (VPC-A)	Peering-AC	自定义	在VPC-C的路由表中，添加目的地址为VPC-A网段，下一跳指向Peering-AC的路由。

说明

路由目的地址配置为对方VPC网段，可以访问对方VPC内的所有资源，VPC网段不能重叠。

配置一个中心 VPC 与两个 VPC 对等（IPv6）

本示例中，在VPC-A和VPC-B之间创建对等连接Peering-AB，在VPC-A和VPC-C之间创建对等连接Peering-AC。VPC-A、VPC-B和VPC-C内的子网都具有IPv6网段，并且VPC-A、VPC-B和VPC-C的IPv4网段不能重叠。

- 资源规划详情，请参见**表11-21**。
- 对等连接关系，请参见**表11-22**。

图 11-7 一个中心 VPC 与两个 VPC 对等(IPv6)

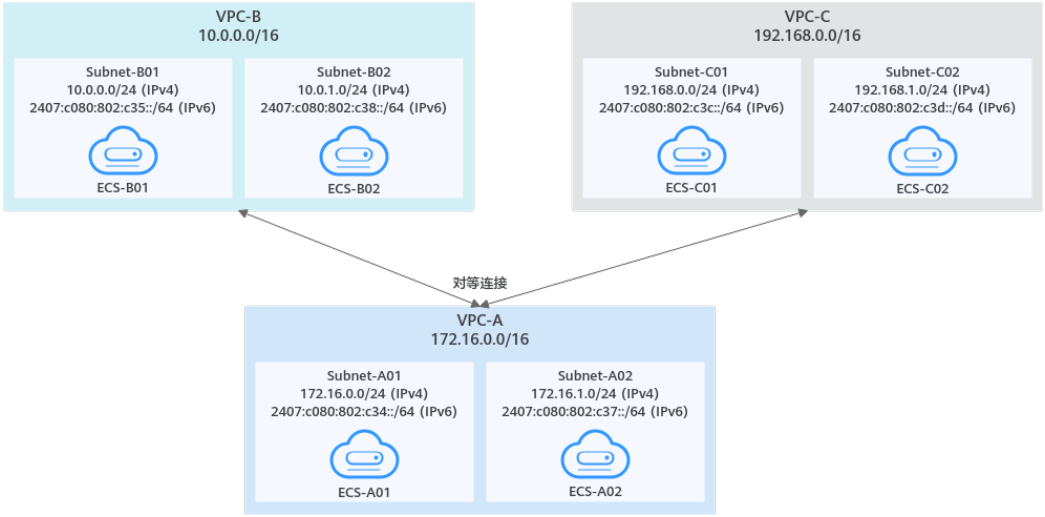


表 11-21 资源规划详情-一个中心 VPC 与两个 VPC 对等(IPv6)

VPC名称	VPC网段	子网名称	子网网段	子网关联VPC路由表	ECS名称	安全组	私有IP地址
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：通用Web服务器	• 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

VPC名称	VPC网段	子网名称	子网网段	子网关联VPC路由表	ECS名称	安全组	私有IP地址
		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-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

表 11-22 对等连接关系说明-一个中心 VPC 与两个 VPC 对等(IPv6)

VPC对等关系	对等连接名称	本端VPC	对端VPC
VPC-A和VPC-B对等	Peering-AB	VPC-A	VPC-B
VPC-A和VPC-C对等	Peering-AC	VPC-A	VPC-C

对等连接创建完成后，您需要在本端和对端VPC路由表中，添加以下路由：

表 11-23 VPC 路由表配置说明-一个中心 VPC 与两个 VPC 对等(IPv6)

路由表	目的地址	下一跳	路由类型	路由说明
rtb-VPC-A	172.16.0.0/24	Local	系统路由	Local路由是系统自动添加的，用于VPC内部通信。
	2407:c080:802:c34::/64	Local	系统路由	
	172.16.1.0/24	Local	系统路由	
	2407:c080:802:c37::/64	Local	系统路由	
	10.0.0.0/16 (VPC-B)	Peering-AB	自定义	在VPC-A的路由表中，添加目的地址为VPC-B网段，下一跳指向Peering-AB的路由，用于IPv4通信。
	2407:c080:802:c35::/64 (Subnet-B01)	Peering-AB	自定义	在VPC-A的路由表中，添加目的地址为子网Subnet-B01和Subnet-B02的IPv6网段，下一跳指向Peering-AB的路由，用于IPv6通信。
	2407:c080:802:c38::/64 (Subnet-B02)	Peering-AB	自定义	
	192.168.0.0/16 (VPC-C)	Peering-AC	自定义	在VPC-A的路由表中，添加目的地址为VPC-C网段，下一跳指向Peering-AC的路由，用于IPv4通信。
	2407:c080:802:c3c::/64 (Subnet-C01)	Peering-AC	自定义	在VPC-A的路由表中，添加目的地址为子网Subnet-C01和Subnet-C02的IPv6网段，下一跳指向Peering-AC的路由，用于IPv6通信。
	2407:c080:802:c3d::/64 (Subnet-C02)	Peering-AC	自定义	
rtb-VPC-B	10.0.0.0/24	Local	系统路由	Local路由是系统自动添加的，用于VPC内部通信。
	2407:c080:802:c35::/64	Local	系统路由	
	10.0.1.0/24	Local	系统路由	
	2407:c080:802:c38::/64	Local	系统路由	

路由表	目的地址	下一跳	路由类型	路由说明
	172.16.0.0/16 (VPC-A)	Peering-AB	自定义	在VPC-B的路由表中，添加目的地址为VPC-A网段，下一跳指向Peering-AB的路由，用于IPv4通信。
	2407:c080:802:c34::/64 (Subnet-A01)	Peering-AB	自定义	在VPC-B的路由表中，添加目的地址为子网Subnet-A01和Subnet-A02的IPv6网段，下一跳指向Peering-AB的路由，用于IPv6通信。
	2407:c080:802:c37::/64 (Subnet-A02)	Peering-AB	自定义	
rtb-VPC-C	192.168.0.0/24	Local	系统路由	Local路由是系统自动添加的，用于VPC内部通信。
	2407:c080:802:c3c::/64	Local	系统路由	
	192.168.1.0/24	Local	系统路由	
	2407:c080:802:c3d::/64	Local	系统路由	
	172.16.0.0/16 (VPC-A)	Peering-AC	自定义	在VPC-C的路由表中，添加目的地址为VPC-A网段，下一跳指向Peering-AC的路由，用于IPv4通信。
	2407:c080:802:c34::/64 (Subnet-A01)	Peering-AC	自定义	在VPC-C的路由表中，添加目的地址为子网Subnet-A01和Subnet-A02的IPv6网段，下一跳指向Peering-AC的路由，用于IPv6通信。
	2407:c080:802:c37::/64 (Subnet-A02)	Peering-AC	自定义	

说明

当前支持通过管理控制台获取VPC子网的IPv6地址，为了实现整个VPC资源互访，此处您需要逐次添加VPC内所有子网的IPv6网段。

配置一个中心 VPC 的主网段和扩展网段与两个 VPC 对等（IPv4）

本示例中，在VPC-A和VPC-B之间创建对等连接Peering-AB，在VPC-A和VPC-C之间创建对等连接Peering-AC，其中VPC-A有主网段和扩展网段，并且VPC-A、VPC-B和VPC-C的网段不能重叠。

- 资源规划详情，请参见**表11-24**。
- 对等连接关系，请参见**表11-25**。

图 11-8 一个中心 VPC 的主网段和扩展网段与两个 VPC 对等(IPv4)

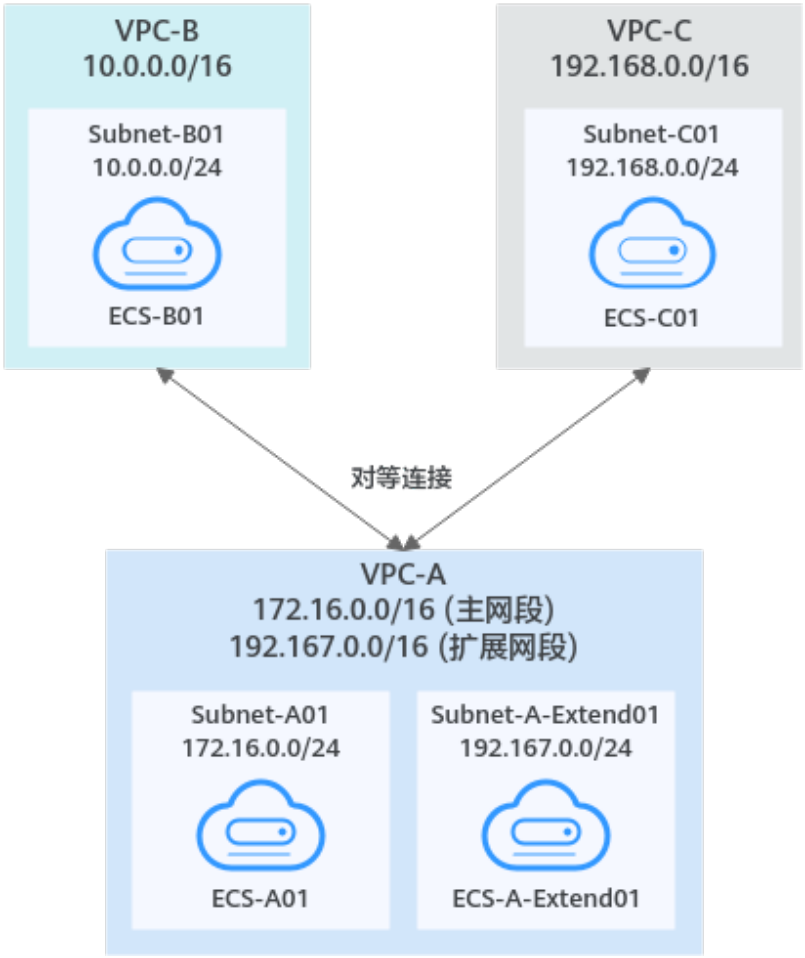


表 11-24 资源规划详情-一个中心 VPC 的主网段和扩展网段与两个 VPC 对等(IPv4)

VPC名称	VPC网段	子网名称	子网网段	子网关联VPC路由表	ECS名称	安全组	私有IP地址
VPC-A	主网段：172.16.0.0/16	Subnet-A01	172.16.0.0/24	rtb-VPC-A	ECS-A01	sg-web：通用Web服务器	172.16.0.111
	扩展网段：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名称	VPC网段	子网名称	子网网段	子网关联VPC路由表	ECS名称	安全组	私有IP地址
VPC-C	192.168.0.0/16	Subnet-C01	192.168.0.0/24	rtb-VPC-C	ECS-C01		192.168.0.194

表 11-25 对等连接关系说明-一个中心 VPC 的主网段和扩展网段与两个 VPC 对等(IPv4)

VPC对等关系	对等连接名称	本端VPC	对端VPC
VPC-A和VPC-B对等	Peering-AB	VPC-A	VPC-B
VPC-A和VPC-C对等	Peering-AC	VPC-A	VPC-C

对等连接创建完成后，您需要在本端和对端VPC路由表中，添加以下路由：

表 11-26 VPC 路由表配置说明-一个中心 VPC 的主网段和扩展网段与两个 VPC 对等(IPv4)

路由表	目的地址	下一跳	路由类型	路由说明
rtb-VPC-A	172.16.0.0/24	Local	系统路由	Local路由是系统自动添加的，用于VPC内部通信。
	192.167.0.0/24	Local	系统路由	
	10.0.0.0/16 (VPC-B)	Peering-AB	自定义	在VPC-A的路由表中，添加目的地址为VPC-B网段，下一跳指向Peering-AB的路由。
	192.168.0.0/16 (VPC-C)	Peering-AC	自定义	在VPC-A的路由表中，添加目的地址为VPC-C网段，下一跳指向Peering-AC的路由。
rtb-VPC-B	10.0.0.0/24	Local	系统路由	Local路由是系统自动添加的，用于VPC内部通信。
	172.16.0.0/16 (VPC-A主网段)	Peering-AB	自定义	在VPC-B的路由表中，添加目的地址为VPC-A主网段和扩展网段，下一跳指向Peering-AB的路由。
	192.167.0.0/16 (VPC-A扩展网段)	Peering-AB	自定义	

路由表	目的地址	下一跳	路由类型	路由说明
rtb-VPC-C	192.168.0.0/24	Local	系统路由	Local路由是系统自动添加的，用于VPC内部通信。
	172.16.0.0/16 (VPC-A主网段)	Peering-AC	自定义	在VPC-C的路由表中，添加目的地址为VPC-A主网段和扩展网段，下一跳指向Peering-AC的路由。
	192.167.0.0/16 (VPC-A扩展网段)	Peering-AC	自定义	

说明

路由目的地址配置为对方VPC网段，可以访问对方VPC内的所有资源，VPC网段不能重叠。

配置一个中心 VPC 与多个 VPC 对等 (IPv4)

本示例中，在VPC-A和VPC-B、VPC-C、VPC-D、VPC-E、VPC-F以及VPC-G之间创建对等连接，并且VPC-A、VPC-B、VPC-C、VPC-D、VPC-E、VPC-F以及VPC-G的网段不能重叠。

- 资源规划详情，请参见**表11-27**。
- 对等连接关系，请参见**表11-28**。

图 11-9 一个中心 VPC 与多个 VPC 对等(IPv4)

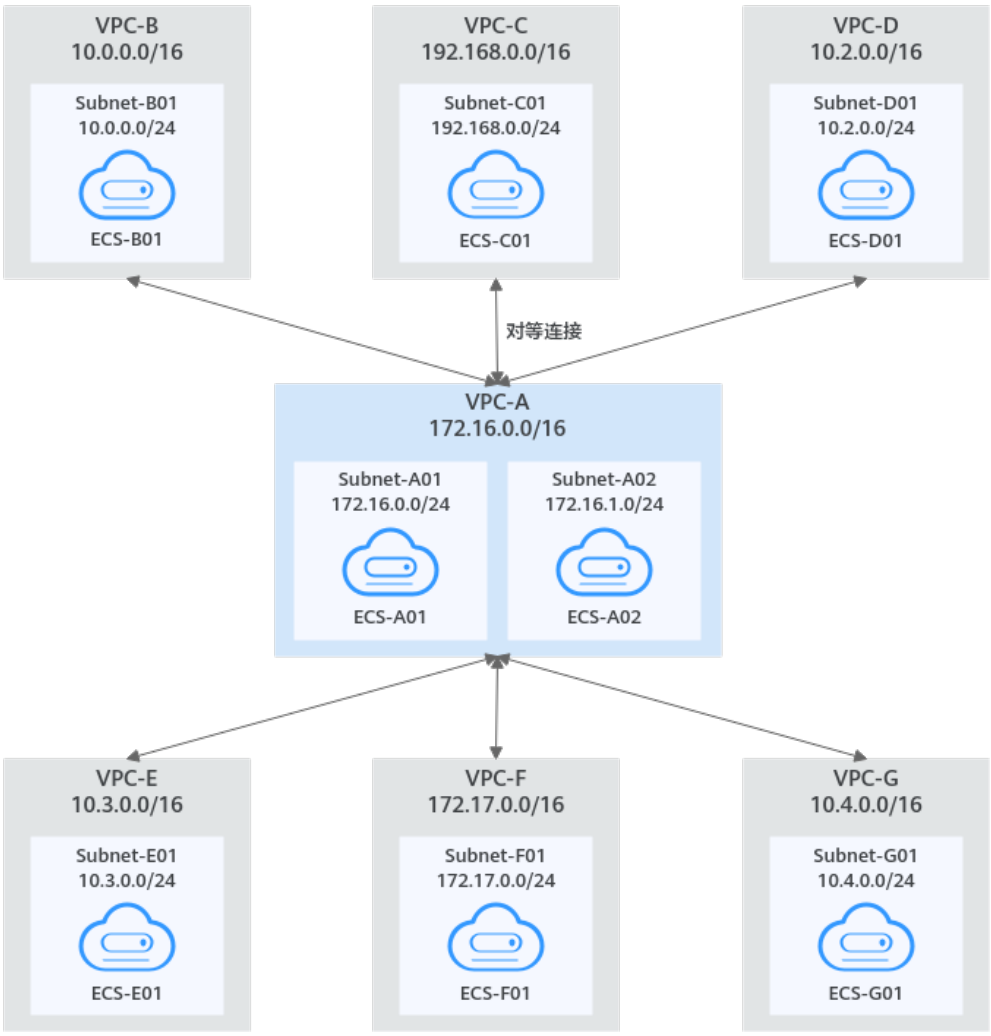


表 11-27 资源规划详情-一个中心 VPC 与多个 VPC 对等(IPv4)

VPC名称	VPC网段	子网名称	子网网段	子网关联VPC路由表	ECS名称	安全组	私有IP地址
VPC-A	172.16.0.0/16	Subnet-A01	172.16.0.0/24	rtb-VPC-A	ECS-A01	sg-web：通用Web服务器	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	192.168.0.0/16	Subnet-C01	192.168.0.0/24	rtb-VPC-C	ECS-C01		192.168.0.194

VPC名称	VPC网段	子网名称	子网网段	子网关联VPC路由表	ECS名称	安全组	私有IP地址
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

表 11-28 对等连接关系说明-一个中心 VPC 与多个 VPC 对等(IPv4)

VPC对等关系	对等连接名称	本端VPC	对端VPC
VPC-A和VPC-B对等	Peering-AB	VPC-A	VPC-B
VPC-A和VPC-C对等	Peering-AC	VPC-A	VPC-C
VPC-A和VPC-D对等	Peering-AD	VPC-A	VPC-D
VPC-A和VPC-E对等	Peering-AE	VPC-A	VPC-E
VPC-A和VPC-F对等	Peering-AF	VPC-A	VPC-F
VPC-A和VPC-G对等	Peering-AG	VPC-A	VPC-G

对等连接创建完成后，您需要在本端和对端VPC路由表中，添加以下路由：

表 11-29 VPC 路由表配置说明-一个中心 VPC 与多个 VPC 对等(IPv4)

路由表	目的地址	下一跳	路由类型	路由说明
rtb-VPC-A	172.16.0.0/24	Local	系统路由	Local路由是系统自动添加的，用于VPC内部通信。
	172.16.1.0/24	Local	系统路由	
	10.0.0.0/16 (VPC-B)	Peering-AB	自定义	在VPC-A的路由表中，添加目的地址为VPC-B网段，下一跳指向Peering-AB的路由。
	192.168.0.0/16 (VPC-C)	Peering-AC	自定义	在VPC-A的路由表中，添加目的地址为VPC-C网段，下一跳指向Peering-AC的路由。
	10.2.0.0/16 (VPC-D)	Peering-AD	自定义	在VPC-A的路由表中，添加目的地址为VPC-D网段，下一跳指向Peering-AD的路由。
	10.3.0.0/16 (VPC-E)	Peering-AE	自定义	在VPC-A的路由表中，添加目的地址为VPC-E网段，下一跳指向Peering-AE的路由。
	172.17.0.0/16 (VPC-F)	Peering-AF	自定义	在VPC-A的路由表中，添加目的地址为VPC-F网段，下一跳指向Peering-AF的路由。
	10.4.0.0/16 (VPC-G)	Peering-AG	自定义	在VPC-A的路由表中，添加目的地址为VPC-G网段，下一跳指向Peering-AG的路由。
rtb-VPC-B	10.0.0.0/24	Local	系统路由	Local路由是系统自动添加的，用于VPC内部通信。
	172.16.0.0/16 (VPC-A)	Peering-AB	自定义	在VPC-B的路由表中，添加目的地址为VPC-A网段，下一跳指向Peering-AB的路由。
rtb-VPC-C	192.168.0.0/24	Local	系统路由	Local路由是系统自动添加的，用于VPC内部通信。
	172.16.0.0/16 (VPC-A)	Peering-AC	自定义	在VPC-C的路由表中，添加目的地址为VPC-A网段，下一跳指向Peering-AC的路由。
rtb-VPC-D	10.2.0.0/24	Local	系统路由	Local路由是系统自动添加的，用于VPC内部通信。
	172.16.0.0/16 (VPC-A)	Peering-AD	自定义	在VPC-D的路由表中，添加目的地址为VPC-A网段，下一跳指向Peering-AD的路由。

路由表	目的地址	下一跳	路由类型	路由说明
rtb-VPC-E	10.3.0.0/24	Local	系统路由	Local路由是系统自动添加的，用于VPC内部通信。
	172.16.0.0/16 (VPC-A)	Peering-AE	自定义	在VPC-E的路由表中，添加目的地址为VPC-A网段，下一跳指向Peering-AE的路由。
rtb-VPC-F	172.17.0.0/24	Local	系统路由	Local路由是系统自动添加的，用于VPC内部通信。
	172.16.0.0/16 (VPC-A)	Peering-AF	自定义	在VPC-F的路由表中，添加目的地址为VPC-A网段，下一跳指向Peering-AF的路由。
rtb-VPC-G	10.4.0.0/24	Local	系统路由	Local路由是系统自动添加的，用于VPC内部通信。
	172.16.0.0/16 (VPC-A)	Peering-AG	自定义	在VPC-G的路由表中，添加目的地址为VPC-A网段，下一跳指向Peering-AG的路由。

说明

路由目的地址配置为对方VPC网段，可以访问对方VPC内的所有资源，VPC网段不能重叠。

配置一个中心 VPC 与多个 VPC 对等 (IPv6)

本示例中，在VPC-A和VPC-B、VPC-C、VPC-D、VPC-E、VPC-F以及VPC-G等之间创建对等连接。VPC-A、VPC-B、VPC-C、VPC-D、VPC-E、VPC-F以及VPC-G内的子网都具有IPv6网段，并且这些VPC的IPv4网段不能重叠。

- 资源规划详情，请参见**表11-30**。
- 对等连接关系，请参见**表11-31**。

图 11-10 一个中心 VPC 与多个 VPC 对等(IPv6)

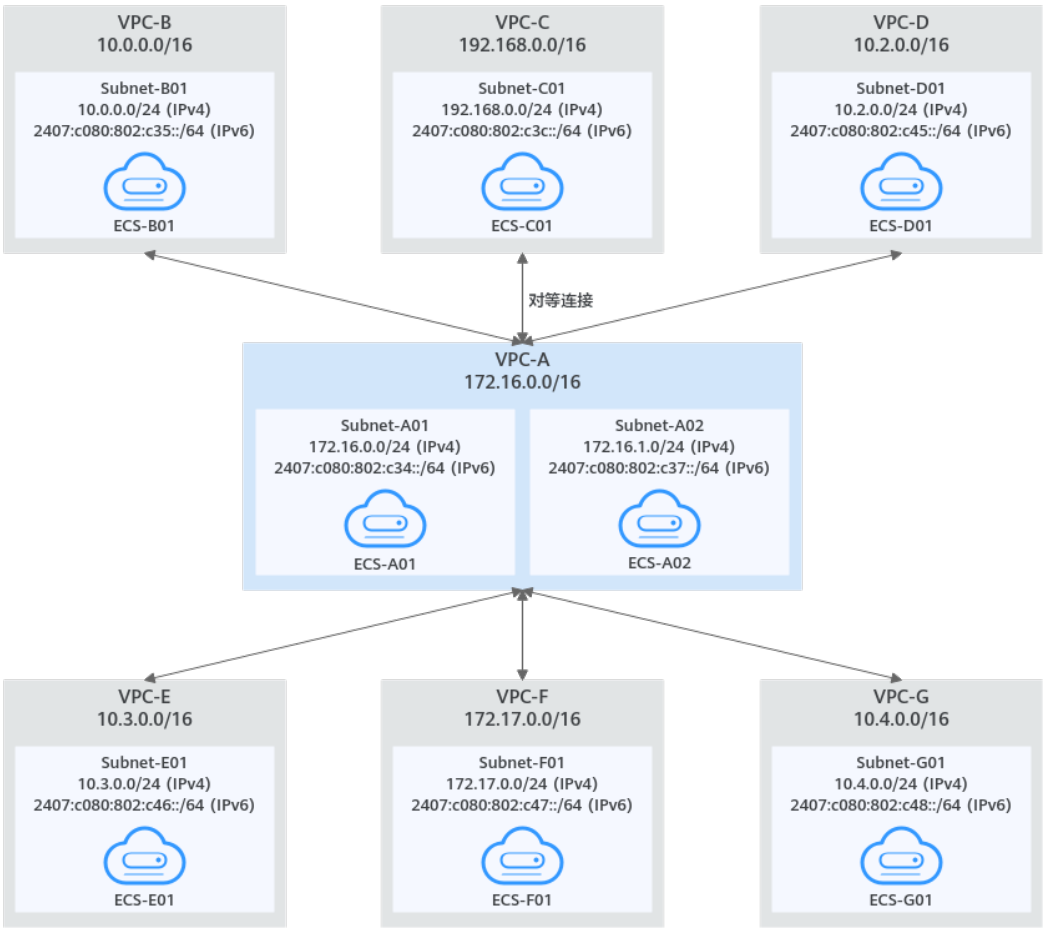


表 11-30 资源规划详情-一个中心 VPC 与多个 VPC 对等(IPv6)

VPC名称	VPC网段	子网名称	子网网段	子网关联VPC路由表	ECS名称	安全组	私有IP地址
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：通用Web服务器	• IPv4: 172.16.0.111 • IPv6: 2407:c080:802:c34:a925:f12e:cfa0:8edb

VPC名称	VPC网段	子网名称	子网网段	子网关联VPC路由表	ECS名称	安全组	私有IP地址
		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名称	VPC网段	子网名称	子网网段	子网关联VPC路由表	ECS名称	安全组	私有IP地址
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

表 11-31 对等连接关系说明-一个中心 VPC 与多个 VPC 对等(IPv6)

VPC对等关系	对等连接名称	本端VPC	对端VPC
VPC-A和VPC-B对等	Peering-AB	VPC-A	VPC-B
VPC-A和VPC-C对等	Peering-AC	VPC-A	VPC-C

VPC对等关系	对等连接名称	本端VPC	对端VPC
VPC-A和VPC-D对等	Peering-AD	VPC-A	VPC-D
VPC-A和VPC-E对等	Peering-AE	VPC-A	VPC-E
VPC-A和VPC-F对等	Peering-AF	VPC-A	VPC-F
VPC-A和VPC-G对等	Peering-AG	VPC-A	VPC-G

对等连接创建完成后，您需要在本端和对端VPC路由表中，添加以下路由：

表 11-32 VPC 路由表配置说明-一个中心 VPC 与多个 VPC 对等(IPv6)

路由表	目的地址	下一跳	路由类型	路由说明
rtb-VPC-A	172.16.0.0/24	Local	系统路由	Local路由是系统自动添加的，用于VPC内部通信。
	2407:c080:802:c34::/64	Local	系统路由	
	172.16.1.0/24	Local	系统路由	
	2407:c080:802:c37::/64	Local	系统路由	
	10.0.0.0/16 (VPC-B)	Peering-AB	自定义	在VPC-A的路由表中，添加目的地址为VPC-B网段，下一跳指向Peering-AB的路由，用于IPv4通信。
	2407:c080:802:c35::/64 (Subnet-B01)	Peering-AB	自定义	在VPC-A的路由表中，添加目的地址为子网Subnet-B01的IPv6网段，下一跳指向Peering-AB的路由，用于IPv6通信。
	192.168.0.0/16 (VPC-C)	Peering-AC	自定义	在VPC-A的路由表中，添加目的地址为VPC-C网段，下一跳指向Peering-AC的路由，用于IPv4通信。
	2407:c080:802:c3c::/64 (Subnet-C01)	Peering-AC	自定义	在VPC-A的路由表中，添加目的地址为子网Subnet-C01的IPv6网段，下一跳指向Peering-AC的路由，用于IPv6通信。
	10.2.0.0/16 (VPC-D)	Peering-AD	自定义	在VPC-A的路由表中，添加目的地址为VPC-D网段，下一跳指向Peering-AD的路由，用于IPv4通信。

路由表	目的地址	下一跳	路由类型	路由说明
	2407:c080:802:c45::/64 (Subnet-D01)	Peering-AD	自定义	在VPC-A的路由表中，添加目的地址为子网Subnet-D01的IPv6网段，下一跳指向Peering-AD的路由，用于IPv6通信。
	10.3.0.0/16 (VPC-E)	Peering-AE	自定义	在VPC-A的路由表中，添加目的地址为VPC-E网段，下一跳指向Peering-AE的路由，用于IPv4通信。
	2407:c080:802:c46::/64 (Subnet-E01)	Peering-AE	自定义	在VPC-A的路由表中，添加目的地址为子网Subnet-E01的IPv6网段，下一跳指向Peering-AE的路由，用于IPv6通信。
	172.17.0.0/16 (VPC-F)	Peering-AF	自定义	在VPC-A的路由表中，添加目的地址为VPC-F网段，下一跳指向Peering-AF的路由，用于IPv4通信。
	2407:c080:802:c47::/64 (Subnet-F01)	Peering-AF	自定义	在VPC-A的路由表中，添加目的地址为子网Subnet-F01的IPv6网段，下一跳指向Peering-AF的路由，用于IPv6通信。
	10.4.0.0/16 (VPC-G)	Peering-AG	自定义	在VPC-A的路由表中，添加目的地址为VPC-G网段，下一跳指向Peering-AG的路由，用于IPv4通信。
	2407:c080:802:c48::/64 (Subnet-G01)	Peering-AG	自定义	在VPC-A的路由表中，添加目的地址为子网Subnet-G01的IPv6网段，下一跳指向Peering-AG的路由，用于IPv6通信。
rtb-VPC-B	10.0.0.0/24	Local	系统路由	Local路由是系统自动添加的，用于VPC内部通信。
	2407:c080:802:c35::/64	Local	系统路由	
	172.16.0.0/16 (VPC-A)	Peering-AB	自定义	在VPC-B的路由表中，添加目的地址为VPC-A网段，下一跳指向Peering-AB的路由，用于IPv4通信。
	2407:c080:802:c34::/64 (Subnet-A01)	Peering-AB	自定义	在VPC-B的路由表中，添加目的地址为子网Subnet-A01和Subnet-A02的IPv6网段，下一跳指向Peering-AB的路由，用于IPv6通信。
	2407:c080:802:c37::/64 (Subnet-A02)	Peering-AB	自定义	
rtb-VPC-C	192.168.0.0/24	Local	系统路由	Local路由是系统自动添加的，用于VPC内部通信。
	2407:c080:802:c3c::/64	Local	系统路由	

路由表	目的地址	下一跳	路由类型	路由说明
	172.16.0.0/16 (VPC-A)	Peering-AC	自定义	在VPC-C的路由表中，添加目的地址为VPC-A网段，下一跳指向Peering-AC的路由，用于IPv4通信。
	2407:c080:802:c34::/64 (Subnet-A01)	Peering-AC	自定义	在VPC-C的路由表中，添加目的地址为子网Subnet-A01和Subnet-A02的IPv6网段，下一跳指向Peering-AC的路由，用于IPv6通信。
	2407:c080:802:c37::/64 (Subnet-A02)	Peering-AC	自定义	
rtb-VPC-D	10.2.0.0/24	Local	系统路由	Local路由是系统自动添加的，用于VPC内部通信。
	2407:c080:802:c45::/64	Local	系统路由	
	172.16.0.0/16 (VPC-A)	Peering-AD	自定义	在VPC-D的路由表中，添加目的地址为VPC-A网段，下一跳指向Peering-AD的路由，用于IPv4通信。
	2407:c080:802:c34::/64 (Subnet-A01)	Peering-AD	自定义	在VPC-D的路由表中，添加目的地址为子网Subnet-A01和Subnet-A02的IPv6网段，下一跳指向Peering-AD的路由，用于IPv6通信。
	2407:c080:802:c37::/64 (Subnet-A02)	Peering-AD	自定义	
rtb-VPC-E	10.3.0.0/24	Local	系统路由	Local路由是系统自动添加的，用于VPC内部通信。
	2407:c080:802:c46::/64	Local	系统路由	
	172.16.0.0/16 (VPC-A)	Peering-AE	自定义	在VPC-E的路由表中，添加目的地址为VPC-A网段，下一跳指向Peering-AE的路由。
	2407:c080:802:c34::/64 (Subnet-A01)	Peering-AE	自定义	在VPC-E的路由表中，添加目的地址为子网Subnet-A01和Subnet-A02的IPv6网段，下一跳指向Peering-AE的路由，用于IPv6通信。
	2407:c080:802:c37::/64 (Subnet-A02)	Peering-AE	自定义	
rtb-VPC-F	172.17.0.0/24	Local	系统路由	Local路由是系统自动添加的，用于VPC内部通信。
	2407:c080:802:c47::/64	Local	系统路由	

路由表	目的地址	下一跳	路由类型	路由说明
	172.16.0.0/16 (VPC-A)	Peering-AF	自定义	在VPC-F的路由表中，添加目的地址为VPC-A网段，下一跳指向Peering-AF的路由。
	2407:c080:802:c34::/64 (Subnet-A01)	Peering-AF	自定义	在VPC-F的路由表中，添加目的地址为子网Subnet-A01和Subnet-A02的IPv6网段，下一跳指向Peering-AF的路由，用于IPv6通信。
	2407:c080:802:c37::/64 (Subnet-A02)	Peering-AF	自定义	
rtb-VPC-G	10.4.0.0/24	Local	系统路由	Local路由是系统自动添加的，用于VPC内部通信。
	2407:c080:802:c48::/64	Local	系统路由	
	172.16.0.0/16 (VPC-A)	Peering-AG	自定义	在VPC-G的路由表中，添加目的地址为VPC-A网段，下一跳指向Peering-AG的路由。
	2407:c080:802:c34::/64 (Subnet-A01)	Peering-AG	自定义	在VPC-G的路由表中，添加目的地址为子网Subnet-A01和Subnet-A02的IPv6网段，下一跳指向Peering-AG的路由，用于IPv6通信。
	2407:c080:802:c37::/64 (Subnet-A02)	Peering-AG	自定义	

说明

当前支持通过管理控制台获取VPC子网的IPv6地址，为了实现整个VPC资源互访，此处您需要逐次添加VPC内所有子网的IPv6网段。

11.3 连通 VPC 子网网络的对等连接配置示例

您可以参考以下示例，配置连通VPC部分子网网络的对等连接，在VPC路由表中添加的路由目的地址为对端VPC子网网段，此时对等连接两端连通的是VPC内指定子网的资源，示例场景如**表11-33**所示。

表 11-33 指向 VPC 子网的对等连接场景说明

组网示例	场景推荐	IP类型	配置示例
两个VPC与一个中心VPC的两个子网对等	当您需要一个中心VPC和其他多个VPC之间资源互访，中心VPC部署的资源类型不同，要求其他多个VPC只能访问中心VPC的特定资源，且彼此之间隔离，可以参考本示例规划组网。 • 中心VPC具有多个子网，不同子网中部署不同类型的资源。 • 其他VPC根据业务需要访问中心VPC内特定子网的资源。	IPv4	配置两个VPC与一个中心VPC的两个子网对等（IPv4）
		IPv6/IPv4	配置两个VPC与一个中心VPC的两个子网对等（IPv6/IPv4）
一个中心VPC与两个VPC的特定子网对等	当您需要一个中心VPC和其他多个VPC之间资源互访，中心VPC部署某类公共资源，其他VPC只有特定子网可以访问中心VPC内的资源，且彼此之间隔离，可以参考本示例规划组网。 • 中心VPC部署的公共资源没有分类，其他VPC可以访问中心VPC内的所有资源。 • 其他VPC具有多个子网，根据业务需要指定某个子网访问中心VPC内的资源。	IPv4	配置一个中心VPC与两个VPC的特定子网对等（IPv4）
一个中心VPC与两个VPC的重叠子网对等	本示例与上面的场景类似，当其他多个VPC和中心VPC对等的子网网段重叠时，可能会导致流量无法被转发到正确的目的地址，请参考本示例规划组网，避免发生该情况。	IPv4	配置一个中心VPC与两个VPC的重叠子网对等（IPv4）

配置两个 VPC 与一个中心 VPC 的两个子网对等（IPv4）

本示例中，中心VPC-A拥有两个子网，并分别关联至不同的路由表。在子网Subnet-A01和VPC-B之间创建对等连接Peering-AB，在子网Subnet-A02和VPC-C之间创建对等连接Peering-AC。此处VPC-B和VPC-C网段重叠，由于VPC-A的两个子网关联至不同的路由表，因此对等连接路由不会冲突。

- 资源规划详情，请参见**表11-34**。
- 对等连接关系，请参见**表11-35**。

图 11-11 两个 VPC 与一个中心 VPC 的两个子网对等(IPv4)

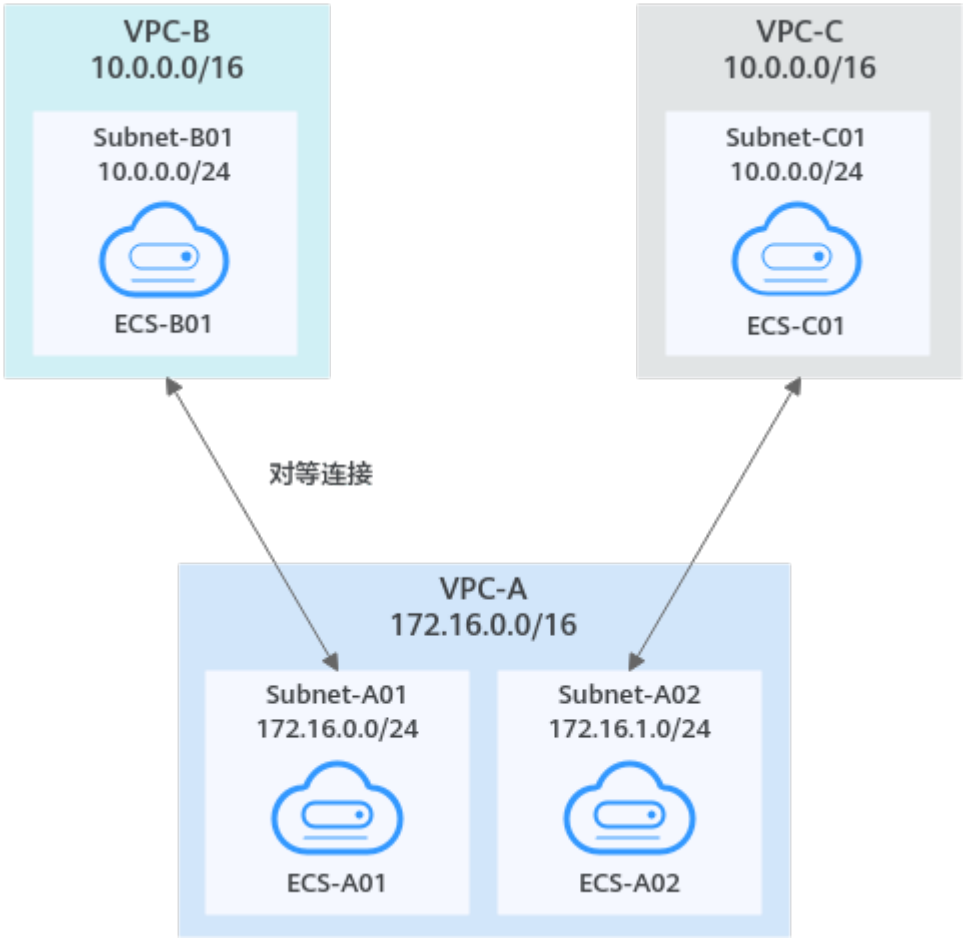


表 11-34 资源规划详情-两个 VPC 与一个中心 VPC 的两个子网对等(IPv4)

VPC名称	VPC网段	子网名称	子网网段	关联VPC路由表	ECS名称	安全组	私有IP地址
VPC-A	172.16.0.0/16	Subnet-A01	172.16.0.0/24	rtb-VPC-A01	ECS-A01	sg-web：通用Web服务器	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

说明

VPC-A有两张路由表，分别关联不同的子网，路由表rtb-VPC-A01关联子网Subnet-A01，路由表子网rtb-VPC-A02关联子网Subnet-A02，两个子网间可正常通信。

表 11-35 对等连接关系说明-两个 VPC 与一个中心 VPC 的两个子网对等(IPv4)

VPC对等关系	对等连接名称	本端VPC	对端VPC
VPC-A的子网Subnet-A01和VPC-B对等	Peering-AB	VPC-A	VPC-B
VPC-A的子网Subnet-A02和VPC-C对等	Peering-AC	VPC-A	VPC-C

对等连接创建完成后，您需要在本端和对端VPC路由表中，添加以下路由：

表 11-36 VPC 路由表配置说明-两个 VPC 与一个中心 VPC 的两个子网对等(IPv4)

路由表	目的地址	下一跳	路由类型	路由说明
rtb-VPC-A01	172.16.0.0/24	Local	系统路由	Local路由是系统自动添加的，用于VPC内部通信。
	172.16.1.0/24	Local	系统路由	
	10.0.0.0/16 (VPC-B)	Peering-AB	自定义	在VPC-A的路由表rtb-VPC-A01中，添加目的地址为VPC-B网段，下一跳指向Peering-AB的路由。
rtb-VPC-A02	172.16.0.0/24	Local	系统路由	Local路由是系统自动添加的，用于VPC内部通信。
	172.16.1.0/24	Local	系统路由	
	10.0.0.0/16 (VPC-C)	Peering-AC	自定义	在VPC-A的路由表rtb-VPC-A02中，添加目的地址为VPC-C网段，下一跳指向Peering-AC的路由。
rtb-VPC-B	10.0.0.0/24	Local	系统路由	Local路由是系统自动添加的，用于VPC内部通信。
	172.16.0.0/24 (Subnet-A01)	Peering-AB	自定义	在VPC-B的路由表中，添加目的地址为Subnet-A01的网段，下一跳指向Peering-AB的路由。
rtb-VPC-C	10.0.0.0/24	Local	系统路由	Local路由是系统自动添加的，用于VPC内部通信。

路由表	目的地址	下一跳	路由类型	路由说明
	172.16.1.0/24 (Subnet-A02)	Peering-AC	自定义	在VPC-C的路由表中，添加目的地址为Subnet-A02的网段，下一跳指向Peering-AC的路由。

配置两个 VPC 与一个中心 VPC 的两个子网对等 (IPv6/IPv4)

本示例中，中心VPC-A拥有两个子网，并分别关联至不同的路由表。在子网Subnet-A01和VPC-B之间创建对等连接Peering-AB，用于IPv6通信。在子网Subnet-A02和VPC-C之间创建对等连接Peering-AC，用于IPv4通信。此处VPC-B和VPC-C网段重叠，由于VPC-A的两个子网关联至不同的路由表，因此对等连接路由不会冲突。

- 资源规划详情，请参见**表11-37**。
- 对等连接关系，请参见**表11-38**。

图 11-12 两个 VPC 与一个中心 VPC 的两个子网对等(IPv6/IPv4)

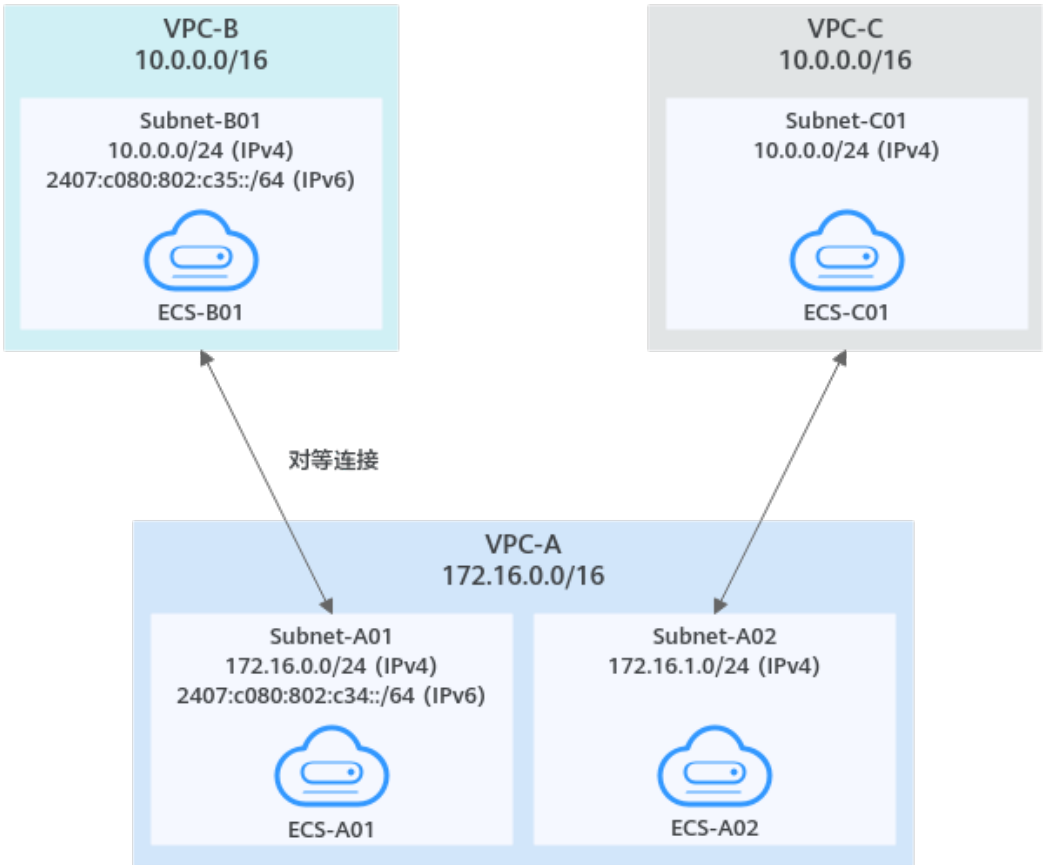


表 11-37 资源规划详情-两个 VPC 与一个中心 VPC 的两个子网对等(IPv6/IPv4)

VPC名称	VPC网段	子网名称	子网网段	关联VPC路由表	ECS名称	安全组	私有IP地址
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：通用Web服务器	• 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-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

说明

VPC-A有两张路由表，分别关联不同的子网，路由表rtb-VPC-A01关联子网Subnet-A01，路由表子网rtb-VPC-A02关联子网Subnet-A02，两个子网间可正常通信。

表 11-38 对等连接关系说明-两个 VPC 与一个中心 VPC 的两个子网对等(IPv6/IPv4)

VPC对等关系	对等连接名称	本端VPC	对端VPC
VPC-A的子网Subnet-A01和VPC-B对等(IPv6)	Peering-AB	VPC-A	VPC-B

VPC对等关系	对等连接名称	本端VPC	对端VPC
VPC-A的子网Subnet-A02和VPC-C对等(IPv4)	Peering-AC	VPC-A	VPC-C

对等连接创建完成后，您需要在本端和对端VPC路由表中，添加以下路由：

表 11-39 VPC 路由表配置说明-两个 VPC 与一个中心 VPC 的两个子网对等(IPv6/IPv4)

路由表	目的地址	下一跳	路由类型	路由说明
rtb-VPC-A01	172.16.0.0/24	Local	系统路由	Local路由是系统自动添加的，用于VPC内部通信。
	2407:c080:802:c34::/64	Local	系统路由	
	172.16.1.0/24	Local	系统路由	
	10.0.0.0/16 (VPC-B)	Peering-AB	自定义	在VPC-A的路由表rtb-VPC-A01中，添加目的地址为VPC-B网段，下一跳指向Peering-AB的路由，用于IPv4通信。
	2407:c080:802:c35::/64 (Subnet-B01)	Peering-AB	自定义	在VPC-A的路由表rtb-VPC-A01中，添加目的地址为子网Subnet-B01的IPv6网段，下一跳指向Peering-AB的路由，用于IPv6通信。
rtb-VPC-A02	172.16.0.0/24	Local	系统路由	Local路由是系统自动添加的，用于VPC内部通信。
	2407:c080:802:c34::/64	Local	系统路由	
	172.16.1.0/24	Local	系统路由	
	10.0.0.0/16 (VPC-C)	Peering-AC	自定义	在VPC-A的路由表rtb-VPC-A02中，添加目的地址为VPC-C网段，下一跳指向Peering-AC的路由，用于IPv4通信。
rtb-VPC-B	10.0.0.0/24	Local	系统路由	Local路由是系统自动添加的，用于VPC内部通信。
	2407:c080:802:c35::/64	Local	系统路由	
	172.16.0.0/24 (Subnet-A01)	Peering-AB	自定义	在VPC-B的路由表中，添加目的地址为Subnet-A01的网段，下一跳指向Peering-AB的路由，用于IPv4通信。

路由表	目的地址	下一跳	路由类型	路由说明
	2407:c080:802:c34::/64 (Subnet-A01)	Peering-AB	自定义	在VPC-B的路由表中，添加目的地址为子网Subnet-A01的IPv6网段，下一跳指向Peering-AB的路由，用于IPv6通信。
rtb-VPC-C	10.0.0.0/24	Local	系统路由	Local路由是系统自动添加的，用于VPC内部通信。
	172.16.1.0/24 (Subnet-A02)	Peering-AC	自定义	在VPC-C的路由表中，添加目的地址为子网Subnet-A02的网段，下一跳指向Peering-AC的路由，用于IPv4通信。

配置一个中心 VPC 与两个 VPC 的特定子网对等 (IPv4)

本示例中，在中心VPC-A和Subnet-B01之间创建对等连接Peering-AB，在中心VPC-A和Subnet-C02之间创建对等连接Peering-AC。此处VPC-B和VPC-C的VPC网段重叠，但是Subnet-B01和Subnet-C02子网网段不重叠，不会造成路由冲突。

- 资源规划详情，请参见**表11-40**。
- 对等连接关系，请参见**表11-41**。

图 11-13 一个中心 VPC 与两个 VPC 的特定子网对等(IPv4)

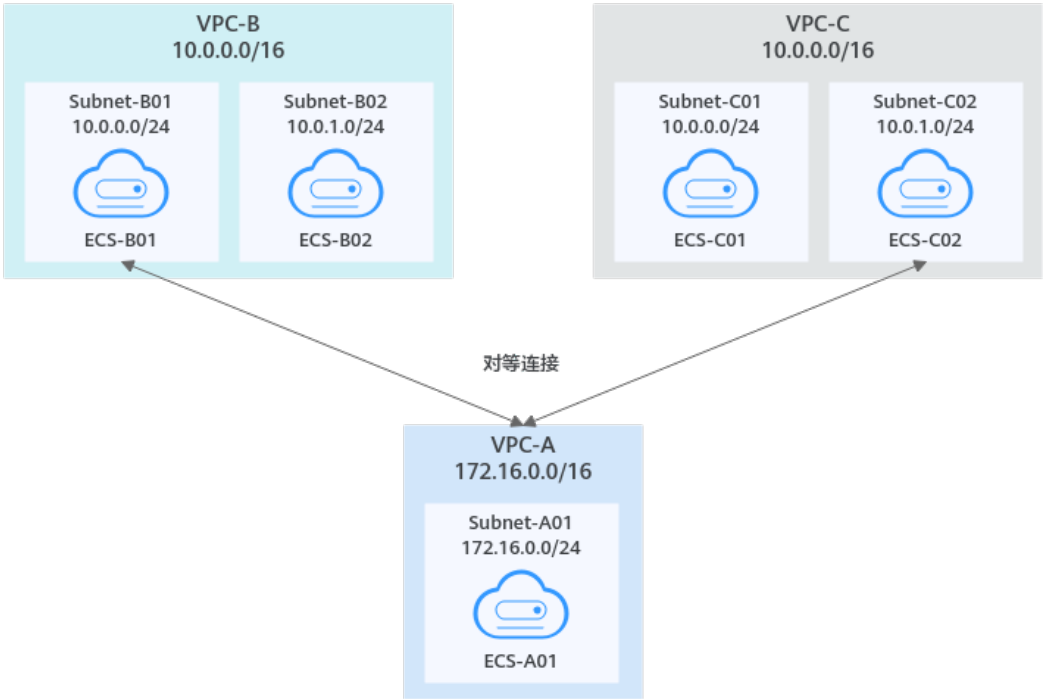


表 11-40 资源规划详情-一个中心 VPC 与两个 VPC 的特定子网对等(IPv4)

VPC名称	VPC网段	子网名称	子网网段	关联VPC路由表	ECS名称	安全组	私有IP地址
VPC-A	172.16.0.0/16	Subnet-A01	172.16.0.0/24	rtb-VPC-A	ECS-A01	sg-web：通用Web服务器	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

表 11-41 对等连接关系说明-一个中心 VPC 与两个 VPC 的特定子网对等(IPv4)

VPC对等关系	对等连接名称	本端VPC	对端VPC
VPC-A和VPC-B的子网Subnet-B01对等	Peering-AB	VPC-A	VPC-B
VPC-A和VPC-C的子网Subnet-C02对等	Peering-AC	VPC-A	VPC-C

对等连接创建完成后，您需要在本端和对端VPC路由表中，添加以下路由：

表 11-42 VPC 路由表配置说明-一个中心 VPC 与两个 VPC 的特定子网对等(IPv4)

路由表	目的地址	下一跳	路由类型	路由说明
rtb-VPC-A	172.16.0.0/24	Local	系统路由	Local路由是系统自动添加的，用于VPC内部通信。
	10.0.0.0/24 (Subnet-B01)	Peering-AB	自定义	在VPC-A的路由表中，添加目的地址为Subnet-B01的网段，下一跳指向Peering-AB的路由。
	10.0.1.0/24 (Subnet-C02)	Peering-AC	自定义	在VPC-A的路由表中，添加目的地址为Subnet-C02的网段，下一跳指向Peering-AC的路由。

路由表	目的地址	下一跳	路由类型	路由说明
rtb-VPC-B	10.0.0.0/24	Local	系统路由	Local路由是系统自动添加的，用于VPC内部通信。
	10.0.1.0/24	Local	系统路由	
	172.16.0.0/16 (VPC-A)	Peering-AB	自定义	在VPC-B的路由表中，添加目的地址为VPC-A的网段，下一跳指向Peering-AB的路由。
rtb-VPC-C	10.0.0.0/24	Local	系统路由	Local路由是系统自动添加的，用于VPC内部通信。
	10.0.1.0/24	Local	系统路由	
	172.16.0.0/16 (VPC-A)	Peering-AC	自定义	在VPC-C的路由表中，添加目的地址为VPC-A的网段，下一跳指向Peering-AC的路由。

配置一个中心 VPC 与两个 VPC 的重叠子网对等 (IPv4)

如果同一个VPC要与多个网段重叠的VPC子网创建对等连接，那么配置路由的时候，请确保路由的目的地址不会出现冲突，并且流量可以正确的转发。

在本示例中，在中心VPC-A和Subnet-B02之间创建对等连接Peering-AB，在中心VPC-A和Subnet-C02之间创建对等连接Peering-AC。此处Subnet-B02和Subnet-C02子网网段重叠，并且云服务器ECS-B02私有IP地址和ECS-C02的私有IP地址一样，均为10.0.1.167/32。

- 资源规划详情，请参见**表11-43**。
- 对等连接关系，请参见**表11-44**。

图 11-14 一个中心 VPC 与两个 VPC 的重叠子网对等(IPv4)

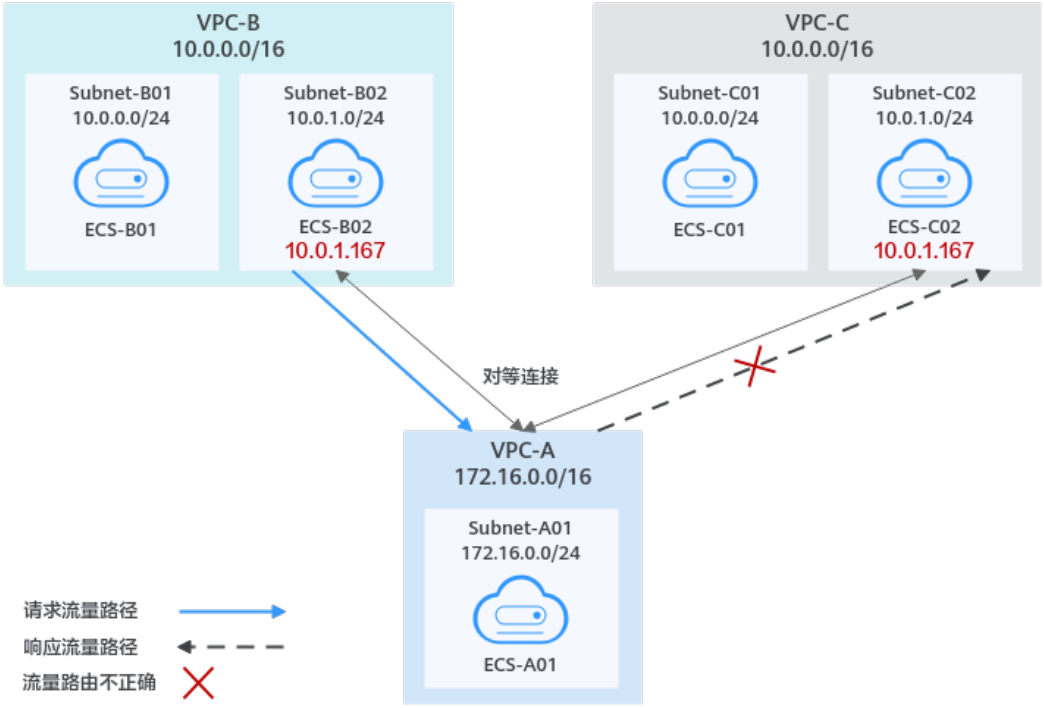


表 11-43 资源规划详情-一个中心 VPC 与两个 VPC 的重叠子网对等(IPv4)

VPC名称	VPC网段	子网名称	子网网段	关联VPC路由表	ECS名称	安全组	私有IP地址
VPC-A	172.16.0.0/16	Subnet-A01	172.16.0.0/24	rtb-VPC-A	ECS-A01	sg-web：通用Web服务器	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

表 11-44 对等连接关系说明-一个中心 VPC 与两个 VPC 的重叠子网对等(IPv4)

VPC对等关系	对等连接名称	本端VPC	对端VPC
VPC-A和VPC-B的子网Subnet-B02对等	Peering-AB	VPC-A	VPC-B

VPC对等关系	对等连接名称	本端VPC	对端VPC
VPC-A和VPC-C的子网Subnet-C02对等	Peering-AC	VPC-A	VPC-C

对等连接创建完成后，您在本端和对端VPC路由表中，如果按照**表11-45**添加路由，那么会导致响应流量无法正确返回，具体说明如下：

1. VPC-B子网Subnet-B02中的云服务器ECS-B02向VPC-A发送请求流量，通过rtb-VPC-B路由表中Peering-AB对应的路由将流量转发到VPC-A。
2. VPC-A收到来自云服务器ECS-B02的请求流量，期望的结果是将响应流量返回到ECS-B02。但是在rtb-VPC-A路由表中，目的地址为10.0.1.167/32时，只能匹配到Peering-AC的路由，因此响应流量被错误的返回到VPC-C。
3. 此时VPC-C的子网Subnet-C02中存在云服务器ECS-C02，与ECS-B02私有IP地址相同，均为10.0.1.167/32，则响应流量最终错误的返回到ECS-C02，ECS-B02无法收到响应流量。

表 11-45 VPC 路由表配置说明-一个中心 VPC 与两个 VPC 的重叠子网对等(IPv4)

路由表	目的地址	下一跳	路由类型	路由说明
rtb-VPC-A	172.16.0.0/24	Local	系统路由	Local路由是系统自动添加的，用于VPC内部通信。
	10.0.1.0/24 (Subnet-C02)	Peering-AC	自定义	在VPC-A的路由表中，添加目的地址为Subnet-C02的网段，下一跳指向Peering-AC的路由。
rtb-VPC-B	10.0.0.0/24	Local	系统路由	Local路由是系统自动添加的，用于VPC内部通信。
	10.0.1.0/24	Local	系统路由	
	172.16.0.0/16 (VPC-A)	Peering-AB	自定义	在VPC-B的路由表中，添加目的地址为VPC-A的网段，下一跳指向Peering-AB的路由。
rtb-VPC-C	10.0.0.0/24	Local	系统路由	Local路由是系统自动添加的，用于VPC内部通信。
	10.0.1.0/24	Local	系统路由	
	172.16.0.0/16 (VPC-A)	Peering-AC	自定义	在VPC-C的路由表中，添加目的地址为VPC-A的网段，下一跳指向Peering-AC的路由。

对于存在重叠子网的情况，为了避免流量被错误转发，我们给您的路由配置建议如下：

- 建议1：在rtb-VPC-A路由表中，添加下一跳为Peering-AB的路由，目的地址为ECS-B02的私有IP地址。这样遵循路由的最长匹配原则，会优先匹配10.0.1.167/32这条路由，确保VPC-A会将响应流量送达ECS-B02。关于更多指向ECS的对等连接配置，请参见**连通VPC内ECS网络的对等连接配置示例**。

表 11-46 VPC 路由表配置-建议 1

路由表	目的地址	下一跳	路由类型	路由说明
rtb-VPC-A	172.16.0.0/24	Local	系统路由	Local路由是系统自动添加的，用于VPC内部通信。
	10.0.1.167/32 (ECS-B02)	Peering-AB	自定义	在VPC-A的路由表中，添加目的地址为ECS-B02的私有IP地址，下一跳指向Peering-AB的路由。
	10.0.1.0/24 (Subnet-C02)	Peering-AC	自定义	在VPC-A的路由表中，添加目的地址为Subnet-C02的网段，下一跳指向Peering-AC的路由。

- 建议2：在rtb-VPC-A路由表中，需要将Peering-AC的路由目的地址由Subnet-C02的网段改为Subnet-C01网段。然后添加下一跳为Peering-AB的路由，目的地址为Subnet-B02，确保VPC-A可以将响应流量返回到VPC-B的子网Subnet-B02。

表 11-47 VPC 路由表配置-建议 2

路由表	目的地址	下一跳	路由类型	路由说明
rtb-VPC-A	172.16.0.0/24	Local	系统路由	Local路由是系统自动添加的，用于VPC内部通信。
	10.0.1.0/24 (Subnet-B02)	Peering-AB	自定义	在VPC-A的路由表中，添加目的地址为Subnet-B02的网段，下一跳指向Peering-AB的路由。
	10.0.0.0/24 (Subnet-C01)	Peering-AC	自定义	在VPC-A的路由表中，添加目的地址为Subnet-C01的网段，下一跳指向Peering-AC的路由。

11.4 连通 VPC 内 ECS 网络的对等连接配置示例

您可以参考以下示例，配置连通VPC内ECS网络的对等连接，在VPC路由表中添加的路由目的地址为ECS的私有IP地址，此时对等连接连通的是不同VPC内的ECS。

当您需要在网段及子网重叠的多个VPC之间创建对等连接时，为了确保路由的正确转发，建议您可以根据组网要求缩小对等连接范围，比如配置不同ECS之间的对等，示例场景如**表11-48**所示。

表 11-48 指向 VPC 内 ECS 的对等连接场景

组网示例	场景推荐	IP类型	配置示例
一个中心VPC的ECS与两个VPC的ECS对等	一个中心VPC和其他两个VPC之间资源互访，其他两个VPC之间隔离。 此场景下，另外两个VPC的网段及子网重叠，此时为了避免中心VPC内的路由冲突，您可以参考本示例，缩小对等连接范围，创建不同VPC内ECS之间的对等连接。	IPv4	**配置一个中心VPC的ECS与两个VPC的ECS对等（IPv4）**
一个中心VPC通过最长匹配原则与两个VPC对等	本示例与上面的场景类似，相比于配置ECS之间的对等连接，您还可以利用路由的最长匹配原则，创建如下两种对等关系： • 在中心VPC和一个VPC的ECS之间创建对等连接 • 在中心VPC和另外一个VPC子网之间创建对等连接 相比ECS对等的组网，该配置方案扩大了对等连接的通信范围。	IPv4	**配置一个中心VPC通过最长匹配原则与两个VPC对等（IPv4）**

配置一个中心 VPC 的 ECS 与两个 VPC 的 ECS 对等（IPv4）

如果在一个VPC和其他多个网段重叠的VPC之间创建对等连接，那么配置路由的时候，您可以参考本示例，将路由的目的地址范围缩小，配置成ECS的私有IP地址。路由目的地址规划不当，会导致流量无法正确转发，错误示例及原因详解，可参见**配置一个中心VPC与两个VPC的重叠子网对等（IPv4）**。

在本示例中，在中心VPC-A内ECS-A01-1和VPC-B内ECS-B01之间创建对等连接Peering-AB，在中心VPC-A内ECS-A01-2和VPC-C内ECS-C01之间创建对等连接Peering-AC。由于Subnet-B01和Subnet-C01子网网段重叠，请确保云服务器ECS-B01和ECS-C01的私有IP地址不同，否则会由于目的地址冲突无法在VPC-A的路由表中添加路由。

- 资源规划详情，请参见**表11-49**。
- 对等连接关系，请参见**表11-50**。

图 11-15 一个中心 VPC 的 ECS 与两个 VPC 的 ECS 对等(IPv4)

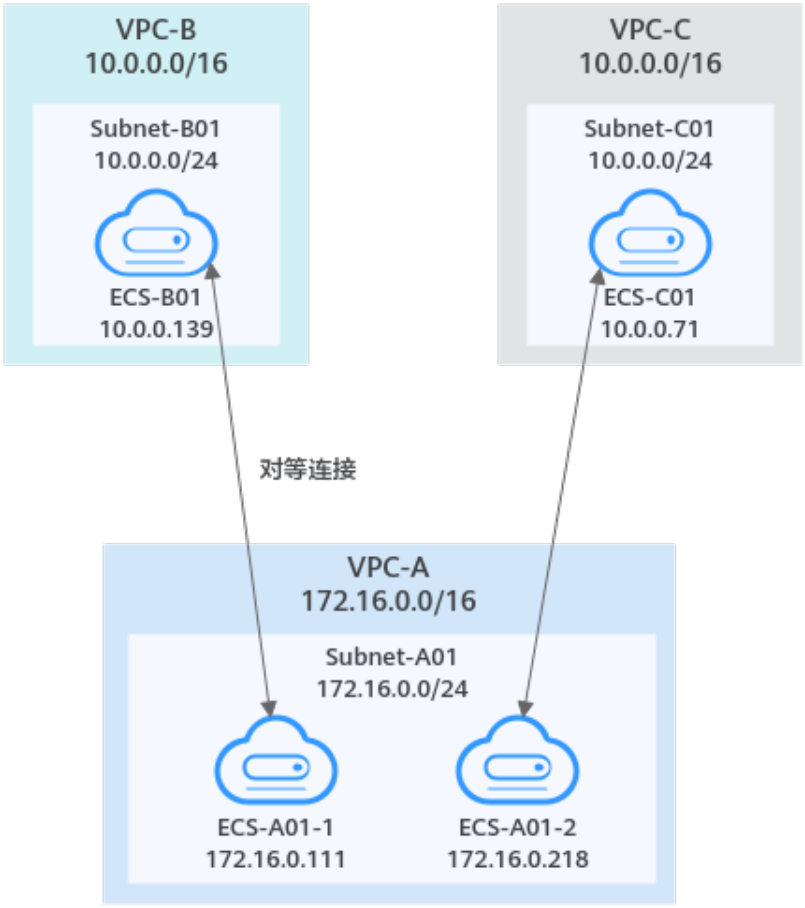


表 11-49 资源规划详情-一个中心 VPC 的 ECS 与两个 VPC 的 ECS 对等(IPv4)

VPC名称	VPC网段	子网名称	子网网段	关联VPC路由表	ECS名称	安全组	私有IP地址
VPC-A	172.16.0.0/16	Subnet-A01	172.16.0.0/24	rtb-VPC-A	ECS-A01-1	sg-web：通用Web服务器	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

表 11-50 对等连接关系说明-一个中心 VPC 的 ECS 与两个 VPC 的 ECS 对等(IPv4)

VPC对等关系	对等连接名称	本端VPC	对端VPC
VPC-A内ECS-A01-1和VPC-B内ECS-B01对等	Peering-AB	VPC-A	VPC-B
VPC-A内ECS-A01-2和VPC-C内ECS-C01对等	Peering-AC	VPC-A	VPC-C

对等连接创建完成后，您在本端和对端VPC路由表中，添加以下路由：

表 11-51 VPC 路由表配置说明-一个中心 VPC 的 ECS 与两个 VPC 的 ECS 对等(IPv4)

路由表	目的地址	下一跳	路由类型	路由说明
rtb-VPC-A	172.16.0.0/24	Local	系统路由	Local路由是系统自动添加的，用于VPC内部通信。
	10.0.0.139/32 (ECS-B01)	Peering-AB	自定义	在VPC-A的路由表中，添加目的地址为ECS-B01的私有IP地址，下一跳指向Peering-AB的路由。
	10.0.0.71/32 (ECS-C01)	Peering-AC	自定义	在VPC-A的路由表中，添加目的地址为ECS-C01的私有IP地址，下一跳指向Peering-AC的路由。
rtb-VPC-B	10.0.0.0/24	Local	系统路由	Local路由是系统自动添加的，用于VPC内部通信。
	172.16.0.111/32 (ECS-A01-1)	Peering-AB	自定义	在VPC-B的路由表中，添加目的地址为ECS-A01-1的私有IP地址，下一跳指向Peering-AB的路由。
rtb-VPC-C	10.0.0.0/24	Local	系统路由	Local路由是系统自动添加的，用于VPC内部通信。
	172.16.0.218/32 (ECS-A01-2)	Peering-AC	自定义	在VPC-C的路由表中，添加目的地址为ECS-A01-2的私有IP地址，下一跳指向Peering-AC的路由。

配置一个中心 VPC 通过最长匹配原则与两个 VPC 对等（IPv4）

如果在一个VPC和其他多个网段重叠的VPC之间创建对等连接，那么配置路由的时候，您可以参考本示例，将路由的目的地址范围缩小，配置成ECS的私有IP地址。路由目的地址规划不当，会导致流量无法正确转发，错误示例及原因详解，可参见**配置一个中心VPC与两个VPC的重叠子网对等（IPv4）**。

在本示例中，在中心VPC-A和VPC-B内ECS-B01之间创建对等连接Peering-AB，在中心VPC-A和VPC-C之间创建对等连接Peering-AC。Subnet-B01和Subnet-C01子网网段重叠，因此添加路由时，可以使用路由的最长匹配原则，控制流量的转发路径。

- 资源规划详情，请参见**表11-52**。
- 对等连接关系，请参见**表11-53**。

图 11-16 一个中心 VPC 通过最长匹配原则与两个 VPC 对等(IPv4)

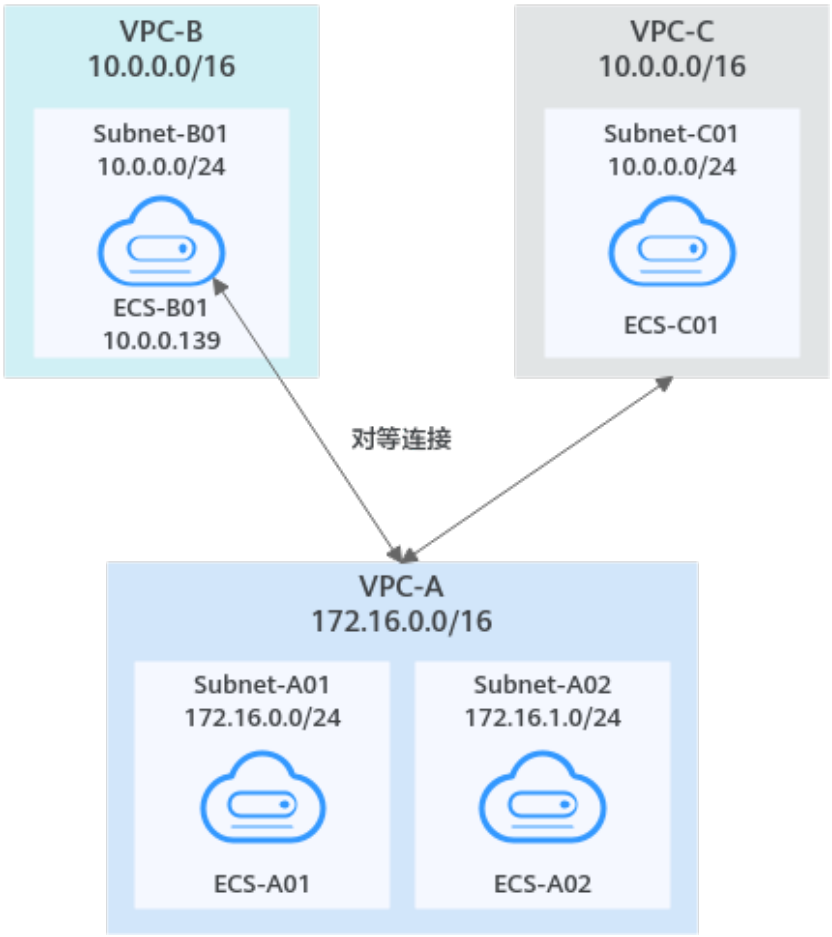


表 11-52 资源规划详情-一个中心 VPC 通过最长匹配原则与两个 VPC 对等(IPv4)

VPC名称	VPC网段	子网名称	子网网段	关联VPC路由表	ECS名称	安全组	私有IP地址
VPC-A	172.16.0.0/16	Subnet-A01	172.16.0.0/24	rtb-VPC-A	ECS-A01	sg-web：通用Web服务器	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名称	VPC网段	子网名称	子网网段	关联VPC路由表	ECS名称	安全组	私有IP地址
VPC-C	10.0.0.0/16	Subnet-C01	10.0.0.0/24	rtb-VPC-C	ECS-C01		10.0.0.71

表 11-53 对等连接关系说明-一个中心 VPC 通过最长匹配原则与两个 VPC 对等(IPv4)

VPC对等关系	对等连接名称	本端VPC	对端VPC
VPC-A和VPC-B内ECS-B01对等	Peering-AB	VPC-A	VPC-B
VPC-A和VPC-C对等	Peering-AC	VPC-A	VPC-C

对等连接创建完成后，您在本端和对端VPC路由表中，添加以下路由：

表 11-54 VPC 路由表配置说明-一个中心 VPC 通过最长匹配原则与两个 VPC 对等(IPv4)

路由表	目的地址	下一跳	路由类型	路由说明
rtb-VPC-A	172.16.0.0/24	Local	系统路由	Local路由是系统自动添加的，用于VPC内部通信。
	172.16.1.0/24	Local	系统路由	
	10.0.0.139/32 (ECS-B01)	Peering-AB	自定义	在VPC-A的路由表中，添加目的地址为ECS-B01的私有IP地址，下一跳指向Peering-AB的路由。
	10.0.0.0/16 (VPC-C)	Peering-AC	自定义	在VPC-A的路由表中，添加目的地址为VPC-C的网段，下一跳指向Peering-AC的路由。
rtb-VPC-B	10.0.0.0/24	Local	系统路由	Local路由是系统自动添加的，用于VPC内部通信。
	172.16.0.0/16 (VPC-A)	Peering-AB	自定义	在VPC-B的路由表中，添加目的地址为VPC-A的网段，下一跳指向Peering-AB的路由。
rtb-VPC-C	10.0.0.0/24	Local	系统路由	Local路由是系统自动添加的，用于VPC内部通信。

路由表	目的地址	下一跳	路由类型	路由说明
	172.16.0.0/16 (VPC-A)	Peering-AC	自定义	在VPC-C的路由表中，添加目的地址为VPC-A的网段，下一跳指向Peering-AC的路由。

11.5 无效的 VPC 对等连接配置示例

操作场景

VPC对等连接针对部分场景的配置是无效的，具体如表11-55所示。

表 11-55 无效的 VPC 对等连接场景

场景说明	配置示例
• VPC网段重叠，且全部子网重叠 此种场景下，不支持使用VPC对等连接。 • VPC网段重叠，且部分子网重叠 此时无法创建指向整个VPC网段的对等连接，可以创建指向子网的对等连接，对等连接两端的子网网段不能包含重叠子网。	**VPC网段重叠可能导致对等连接不生效** • **VPC网段重叠，且全部子网重叠** • **VPC网段重叠，且部分子网重叠**
基于VPC对等连接，无法实现多个ECS共用EIP访问公网。 比如，在VPC-A和VPC-B之间创建对等连接，VPC-A内的云服务器ECS-A01绑定了EIP用来访问公网，此时VPC-B内的云服务器ECS-B01无法通过ECS-A01的EIP访问公网。	**VPC对等连接不支持共用EIP**

VPC 网段重叠可能导致对等连接不生效

VPC网段重叠的情况下，容易因为路由冲突导致对等连接不生效，以下为您提供原因说明和配置建议，请根据您的VPC资源情况进行选择：

- VPC网段重叠，且全部子网重叠

  此场景下，不支持使用对等连接。如图11-17所示，以网段和子网完全重叠的VPC-A和VPC-B为例，假如在VPC-A和VPC-B之间创建对等连接，那么路由表如表11-56所示。

  在rtb-VPC-A路由表中，Local路由和对等连接路由的目的地址重叠，VPC-A往VPC-B的流量，会优先匹配Local路由，流量在VPC-A内部转发，无法送达VPC-B。

图 11-17 VPC 网段重叠，且全部子网重叠(IPv4)

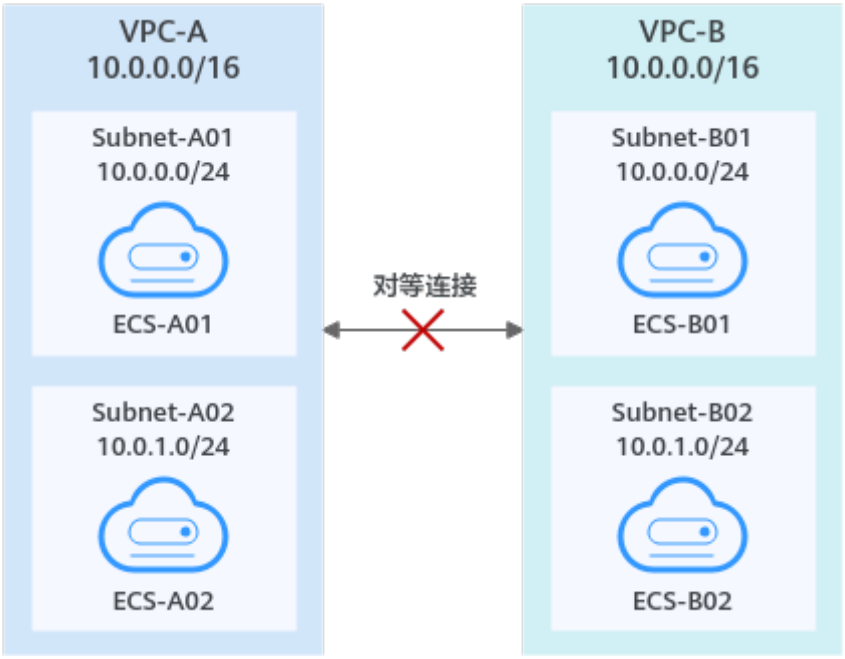
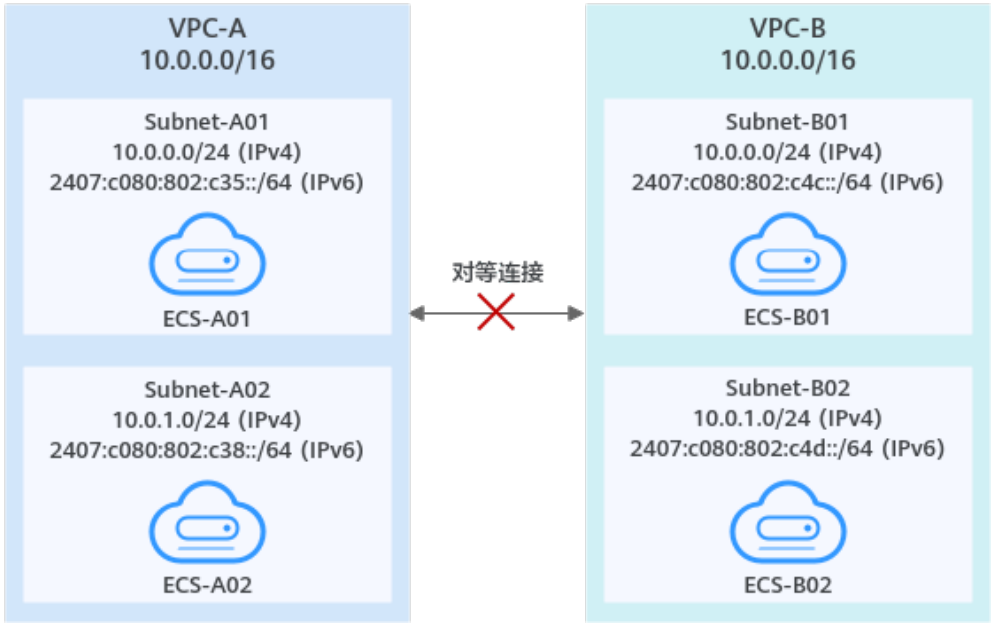


表 11-56 VPC 路由表配置说明-VPC 网段重叠，且全部子网重叠(IPv4)

路由表	目的地址	下一跳	路由类型	路由说明
rtb-VPC-A	10.0.0.0/24	Local	系统路由	Local路由是系统自动添加的，用于VPC内部通信。
	10.0.1.0/24	Local	系统路由	
	10.0.0.0/16 (VPC-B)	Peering-AB	自定义	在VPC-A的路由表中，添加目的地址为VPC-B的网段，下一跳指向Peering-AB的路由。
rtb-VPC-B	10.0.0.0/24	Local	系统路由	Local路由是系统自动添加的，用于VPC内部通信。
	10.0.1.0/24	Local	系统路由	
	10.0.0.0/16 (VPC-A)	Peering-AB	自定义	在VPC-B的路由表中，添加目的地址为VPC-A的网段，下一跳指向Peering-AB的路由。

以上限制同样适用于IPv6场景，即使您只需要使用对等连接实现不同VPC之间的IPv6通信，此时如果对等连接两端VPC的IPv4网段和子网重叠，那么您创建的对等连接也不会生效。

图 11-18 VPC 网段重叠，且全部子网重叠(IPv6)



- VPC网段重叠，且部分子网重叠

  创建VPC对等连接时，如果两端的VPC网段和部分子网重叠，那么请您避免创建以下场景的对等连接：

  - 指向整个VPC网段的对等连接将不生效。

    如图11-19所示，假如创建VPC-A和VPC-B之间的对等连接，由于VPC-A和VPC-B网段重叠，那么对等连接将不生效。

  - 指向VPC子网的对等连接，如果对等连接两端包含重叠子网，将不会生效。

    如图11-19所示，创建Subnet-A01和Subnet-B02之间的对等连接，那么路由表如表11-57所示。在rtb-VPC-B路由表中，Local路由和对等连接的路由目的地址重叠，Subnet-B02往Subnet-A01的流量，会优先匹配Local路由，流量在Subnet-B02内部转发，无法送达Subnet-A01。

图 11-19 VPC 网段重叠，且部分子网重叠(IPv4)

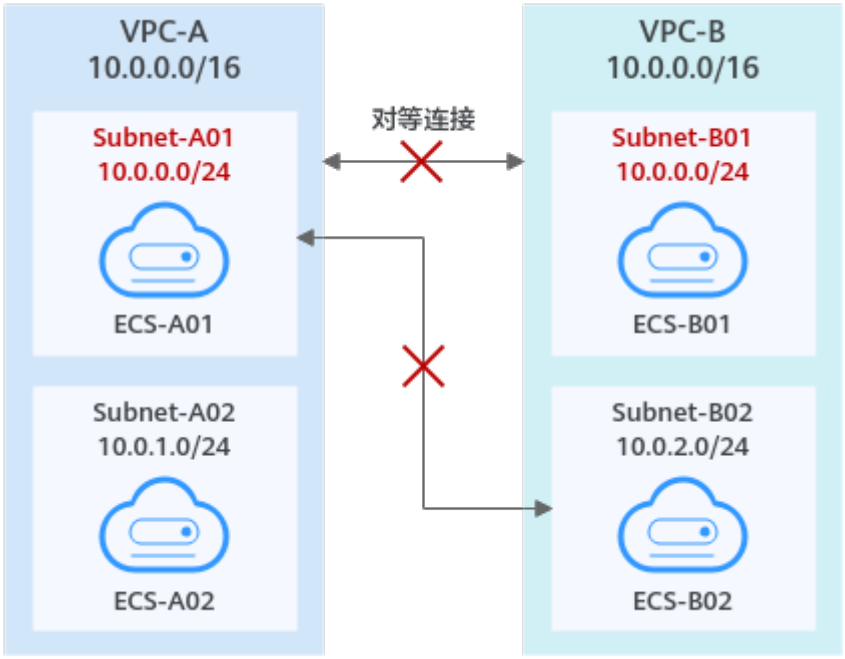
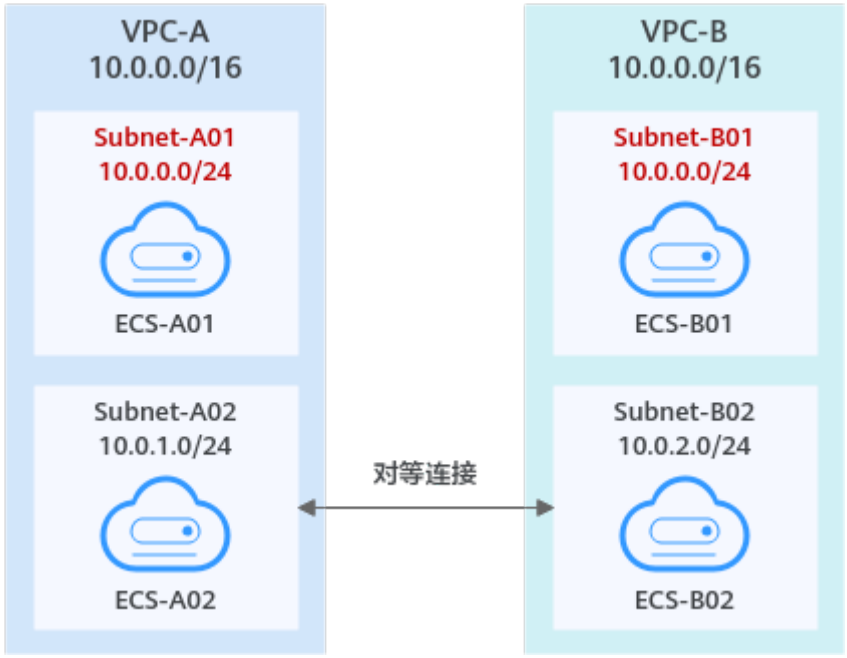


表 11-57 VPC 路由表配置说明-VPC 网段重叠，且部分子网重叠(IPv4)

路由表	目的地址	下一跳	路由类型	路由说明
rtb-VPC-A	10.0.0.0/24	Local	系统路由	Local路由是系统自动添加的，用于VPC内部通信。
	10.0.1.0/24	Local	系统路由	
	10.0.2.0/24 (Subnet-B02)	Peering-AB	自定义	在VPC-A的路由表中，添加目的地址为Subnet-B02子网网段，下一跳指向Peering-AB的路由。
rtb-VPC-B	10.0.0.0/24	Local	系统路由	Local路由是系统自动添加的，用于VPC内部通信。
	10.0.2.0/24	Local	系统路由	
	10.0.0.0/24 (Subnet-A01)	Peering-AB	自定义	在VPC-B的路由表中，添加目的地址为Subnet-A01子网网段，下一跳指向Peering-AB的路由。

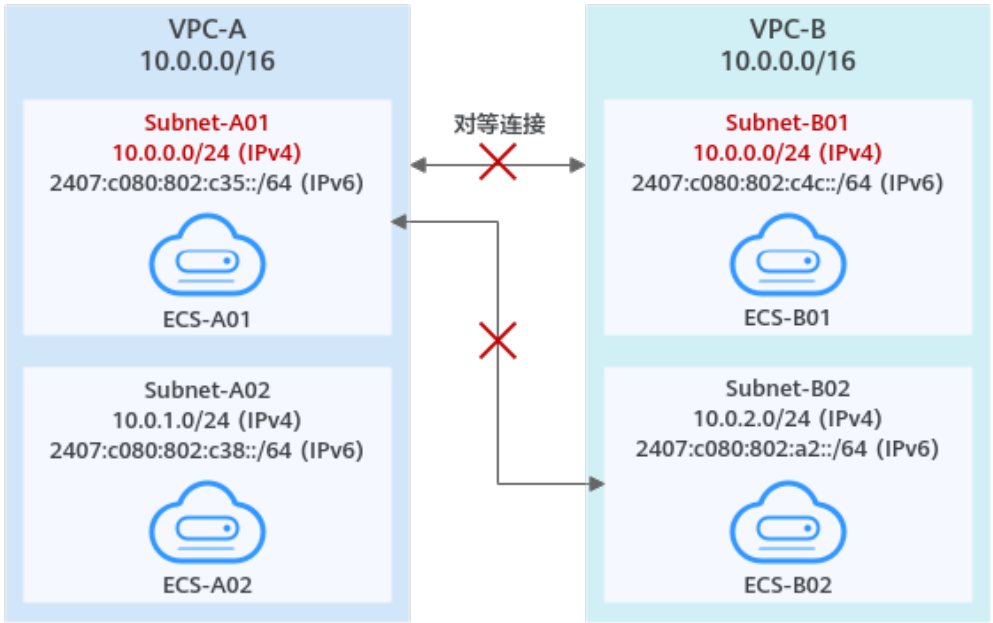
对于此场景，对等连接两端不能包含重叠子网。如图11-20所示，您可以创建Subnet-A02和Subnet-B02之间的对等连接，此时路由不会冲突，对等连接生效。

图 11-20 VPC 网段重叠，部分子网重叠(IPv4)-正确配置



以上限制同样适用于IPv6场景，即使您只需要使用对等连接实现不同VPC之间的IPv6通信，此时如果对等连接两端VPC的IPv4网段和子网重叠，那么您创建的对等连接也不会生效。

图 11-21 VPC 网段重叠，且部分子网重叠(IPv6)



VPC 对等连接不支持共用 EIP

如图11-22所示，在VPC-A和VPC-B之间建立对等连接，ECS-A01绑定了EIP用来访问公网，此时ECS-B01无法通过ECS-A01的EIP访问公网。

图 11-22 VPC 对等连接不支持共用 EIP