



天翼云·物理机服务

用户使用指南

天翼云科技有限公司

目录

1	产品介绍	5
1.1	什么是天翼云物理机服务	5
1.2	天翼云物理机的功能	7
1.3	天翼云物理机的优势	7
1.4	天翼云物理机的可用分区	10
1.4.1	区域和可用区的概念	11
1.4.2	如何选择区域?	11
1.4.3	如何选择可用区?	12
1.5	天翼云物理机的相关存储类型	12
1.6	天翼云物理机的相关网络	13
1.6.1	虚拟私有云	13
1.6.2	高速网络	13
1.6.3	自定义 VLAN 网络	14
1.6.4	IB 网络	14
1.7	物理机实例规格说明	15
1.7.1	实例概述	15
1.7.2	x86 V4 实例 (CPU 采用 Intel Broadwell 架构)	15
1.7.3	x86 V5 实例 (CPU 采用 Intel Skylake 架构)	16
1.7.4	X86 V6 实例 (CPU 采用 Intel Cascade Lake 架构)	19
1.8	天翼云物理机的镜像	20
1.8.1	镜像概述	20
1.8.2	物理机服务器类型与支持的操作系统版本	21
1.9	天翼云物理机的安全	24
1.9.1	安全组	24
1.9.2	密钥对与密码	25
1.10	天翼云物理机的使用场景	26
1.11	天翼云物理机的访问和使用	26
1.12	天翼云物理机的使用限制	29
2	购买指南	31
2.1	物理机规格和价格	31
2.2	物理机计费项	36
2.3	物理机购买	39
2.4	物理机续订	47

2.5	物理机退订	49
2.6	物理机到期处理	50
3	快速入门	51
3.1	入门指引	51
3.2	准备工作	52
3.3	开通物理机服务	53
3.3.1	自助开通	53
3.3.2	通过客户经理开通	59
3.4	查看物理机详情	60
3.5	登录物理机服务	61
3.5.1	登录 linux 物理机	61
3.5.2	登录 windows 物理机	70
3.5.3	初始化数据盘	71
3.5.4	安装一键式重置密码插件	78
3.6	部署应用服务	84
3.7	退订物理机服务	85
4	操作指南	87
4.1	物理机实例管理	87
4.1.1	物理机生命周期管理	87
4.1.2	创建物理机	88
4.1.3	查看物理机信息	95
4.1.4	重置物理机的密码	98
4.1.5	重装物理机操作系统	99
4.2	制作物理机镜像	103
4.2.1	私有镜像概述	103
4.2.2	通过物理机服务器创建私有镜像	103
4.2.3	通过外部镜像文件创建私有镜像	105
4.3	存储	105
4.3.1	挂载磁盘	105
4.3.2	卸载磁盘	106
4.3.3	扩容磁盘	107
4.4	网络	107
4.4.1	物理机网络介绍	107
4.4.2	VPC 网络	109

4.4.3	高速网络.....	110
4.4.4	自定义 VLAN 网络.....	111
4.5	密钥与密码.....	112
4.6	管理弹性 IP.....	117
4.6.1	查看弹性 IP.....	117
4.6.2	绑定弹性 IP.....	117
4.6.3	解绑弹性 IP.....	118
4.7	配置安全组.....	118
4.7.1	添加安全组规则.....	118
4.7.2	物理机器的安全组配置案例.....	119
4.8	管理虚拟 IP 地址.....	122
4.9	设置网卡的源/目的检查.....	122
4.10	管理高速网络.....	123
4.10.1	使用限制.....	123
4.10.2	创建高速网络.....	123
4.10.3	修改高速网络名称.....	124
4.10.4	管理私有 IP 地址.....	124
4.11	自定义 VLAN 网络配置.....	125
4.11.1	SUSE Linux Enterprise Server 12 系列.....	158
4.11.2	SUSE Linux Enterprise Server 11 系列.....	167
4.11.3	RedHat 系列, CentOS 系列, Oracle Linux 系列, Euler 系列.....	174
4.11.4	Ubuntu 系列.....	179
4.11.5	Windows Server 系列.....	185
4.12	物理机监控.....	190
4.12.1	物理机监控概述.....	204
4.12.2	安装配置 Agent.....	205
4.12.3	物理机支持的监控指标.....	209
4.13	故障排除.....	211
4.13.1	物理机开机或者重启后, 无法登录或 EVS 磁盘丢失, 如何解决?.....	211
4.13.2	通过 puttygen.exe 工具创建的密钥对, 导入管理控制台失败怎么办?.....	211
4.13.3	物理机异常重启后无法挂卷, 如何处理?.....	214
4.13.4	Windows 物理机挂载云硬盘后提示脱机, 如何解决?.....	214
5	常见问题.....	216
	常见问题.....	216

5.1.1	通用类	216
5.1.2	计费类	218
5.1.3	购买类	220
5.1.4	管理类	221
5.1.5	登录类	221
5.1.6	网络与安全类.....	226
5.1.7	磁盘类	227
5.1.8	操作系统类.....	233

1 产品介绍

1.1 什么是天翼云物理机服务

本章节主要介绍了什么是物理机和物理机产品架构,以及与自建物理机、云主机的功能对比。

产品定义:

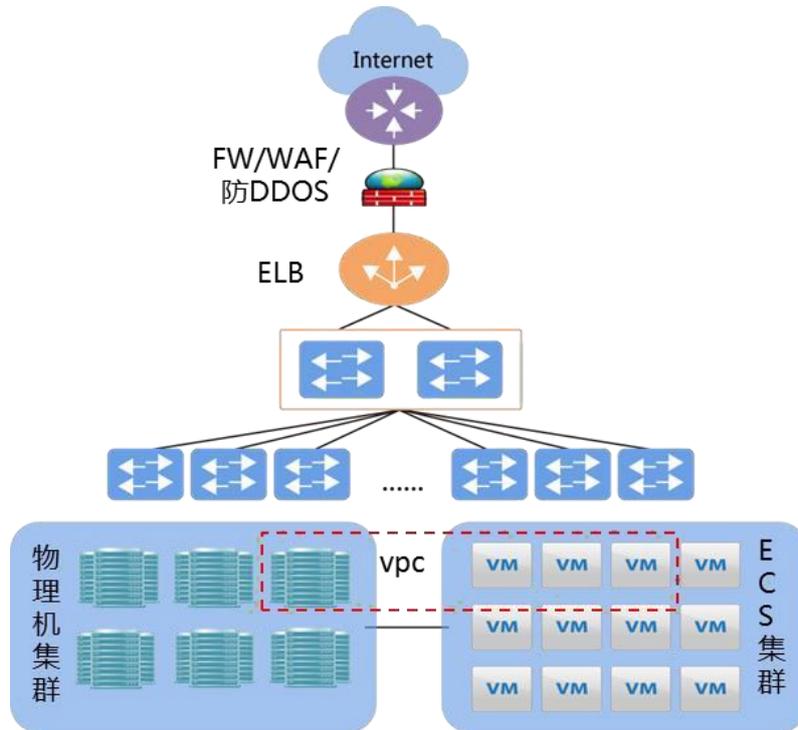
天翼云物理机服务 (CT-DPS, Dedicated Physical Server) 是一款兼具弹性云主机和物理机性能的计算类服务,为您及您的企业提供专属独享的物理机服务,为核心数据库、关键应用系统、高性能计算、大数据等业务提供卓越的计算性能以及数据安全,满足核心应用对高性能及稳定性的需求。同时,可实现与弹性云主机混合组网,为用户提供灵活的业务部署方案。

产品架构:

天翼云物理机通过和其他服务组合,可以实现计算、存储、网络、镜像安装等功能:

- 在不同可用区中部署 (可用区之间通过内网连接) 物理机,部分可用区发生故障后不会影响同一区域内的其他可用区。
- 可以通过虚拟私有云建立专属的网络环境,设置子网、安全组,并通过弹性公网 IP 实现外网链接。
- 通过镜像服务,可以对物理机安装镜像,也可以通过私有镜像批量创建物理机,实现快速部署业务。
- 通过云硬盘服务实现数据存储,并通过云硬盘备份服务实现数据的备份和恢复。
- 云监控是保持物理机服务器可靠性、可用性和性能的重要部分,通过云监控,用户可以观察物理机服务器资源。
- 云备份提供对云硬盘和物理机的备份保护服务,支持基于快照技术的备份服务,并支持利用备份数据恢复服务器和磁盘的数据。

产品架构图1:



天翼云物理机与自建物理机、云主机的功能对比

天翼云物理机与自建物理机、云主机的对比如下表 1 所示。其中，√表示支持，×表示不支持。

表 1 特性对比:

功能分类	功能	天翼云物理机	自建物理机	云主机
发放方式	自动化发放	√	×	√
计算	无特性损失	√	√	×
	无性能损失	√	√	×
	无资源争抢	√	√	×
存储	拥有本地存储	√	√	×
	使用云硬盘（系统盘）	√	×	√

	启动			
	使用镜像，免操作系统安装	√	×	√
网络	使用虚拟私有云网络	√	×	√
	物理机集群和云主机集群之间通过 VPC 通信	√	×	√
管控	VNC 远程登录等体验和云主机一致	√	×	√
	支持监控以及关键操作审计	√	×	√

1.2 天翼云物理机的优势

本章节主要介绍天翼云物理机的优势。

成本优越

- **天翼云物理机：**以租用的方式提供，与公有云产品无缝连接，高可用，易扩展，只需支付实例费用，免安装免运维，无需投入大量资金。
- **传统 IDC 物理机：**功能单一，扩展性差，需投入大量人力成本和物料成本，包括服务器、系统、数据库等软硬件费用，及机房机柜费用和运维成本。

性能卓越

- 采用市场主流硬件设备，并继承传统物理服务器特性，为用户提供专属的物理机稳定性，无虚拟化开销和性能损失，100%释放算力资源。
- 结合天翼云全新分布式 4.0 架构，基于深度学习智能调度引擎，超大规模部署，支持高带宽、低时延云存储、云网络访问性能，带宽最高可达 10Gbits，时延低至 25us 等；满足企业数据库、大数据、容器、HPC、AI 等关键业务部署密度和性能诉求。

安全可靠

- 不同用户全面资源隔离，用户独享计算资源，数据安全有保障
- 支持 VPC、安全组隔离；支持主机安全相关组件集成
- 支持云磁盘作为系统盘和数据盘，支持硬盘备份恢复能力；支持对接专属存储，满足企业数据安全和监管的业务安全和可靠性诉求。

高效部署

- 支持云磁盘作为系统盘快速发放；分钟级发放物理机，不受资源约束，满足用户对资源弹性和快速部署需求
- 基于统一 console 控制台，支持自助式资源生命周期管理和运维管理。

快速响应

- 专业的运维团队，硬件、网络故障将被直接处理，快速响应
- 监控可视化，配件齐全，一旦出现故障及时替换，保证用户可用性

灵活组合

- 支持与云主机混合组网，操作简单，满足不同业务部署需求。

高可靠存储

- 采用云磁盘作为数据持久化，支持服务器重建，整体可用性、可靠性高于传统服务器
- 支持在线挂载、卸载云硬盘，满足用户对弹性存储的诉求
- 支持云磁盘作为系统盘和数据盘、硬盘备份恢复能力
- 支持共享磁盘，更好地满足搭建集群高可用场景的应用

1.3 天翼云物理机的功能

本章节主要介绍天翼云物理机的功能。

订购及管理

物理机服务的开通/订购需通过联系客服或客户经理的方式，由天翼云后台进行开通处理，

并可在控制台对物理机进行开机、关机、重启等管理操作。

提供多种操作系统

支持常用的 Windows、Linux 操作系统，用户订购物理机时可指定操作系统类型。

物理机与云主机网络互通

物理机服务支持部署在 VPC 中，支持 IPv6 特性，可实现物理机与物理机之间、物理机与云主机之间网络互通，满足不同业务场景对物理机、云主机部署的需求。

物理机支持多种网络类型：VPC 网络、高速网络、IB 网络、自定义 VLAN 网络等。

物理机与云存储

支持挂载云硬盘，在线扩容，解决了传统物理服务器受限于本地硬盘容量的问题。

支持挂载共享云硬盘，由多台服务器并发读写访问，满足企业核心系统集群部署的需求。

物理机镜像

物理机支持公共镜像、私有镜像和共享镜像。用户可通过选择公共镜像快速发放物理机，免操作系统安装，通过定制的私有镜像可以一键创建所需要的物理机服务器，节省重复配置物理机的时间。

物理机生命周期管理

支持物理机和 HPC 生命周期管理，可以通过公有镜像或者私有镜像进行快速部署及发放，网络自动配置，发放完成即可使用。

混合部署，灵活组网

物理机在可用分区内，内网互通。通过 VPC 实现与外部资源的互通，同时可以结合 ECS 等服务混合部署、灵活组网，满足用户多种复杂场景的不同诉求。

稳定可靠，性能卓越

为租户提供专属的物理机，独享物理机的稳定性能，无虚拟化开销和性能损失，100%释放算力资源。充分满足对高性能、稳定性以及数据安全和监管的业务诉求。

高吞吐量，低时延

可为租户提供同一可用区内物理机之间高吞吐、高带宽、低时延网络，带宽最高可达 10Gbit。

1.4 物理机的使用场景

本章节主要介绍天翼云物理机的使用场景。

对安全和监管高要求的场景

金融、证券等行业对业务部署的合规性，以及某些客户对数据安全有苛刻的要求。采用物理机部署，通过网络隔离、专线接入等方式确保独享资源，数据安全隔离，保障用户数据的安全。

核心数据库

泛政务、金融、互联网等行业客户的关键业务系统的核心数据库，其业务压力大、安全隔离要求高，可通过资源专享、网络隔离、性能有保障的物理机承载，满足业务需求。尤其类似 Oracle 等数据库，其部署复杂，对性能要求高，推荐采用物理机部署。

大数据场景

互联网、汽车、交通、政务等行业的大数据存储、大数据分析等相关业务，客户在云上自建大数据平台，推荐采用性能更强、兼容性更高的物理机服务器承载。同时，天翼云物理机服务器支持结合对象存储服务的存算分离方案。

容器场景

电商平台、游戏、疫情核酸平台等业务弹性要求较高，性能要求更高的场景，可采用物理机容器方案，相比虚拟机容器，物理机容器提供更高的部署密度、更低的资源开销、更加敏捷的部署效率。基于云原生技术帮助客户实现降低云化成本目标。

高性能计算

科研、基因测序、天气预测、能源勘探、影视渲染、人工智能等高性能计算场景下，对计算能力要求高、处理的任务多、并行数据量大。天翼云物理机采用高规格物理服务与本地 SAS 存储进行部署，提供强大的计算能力和存储性能。高计算性能、稳定性和实时性，避免虚拟化带来的性能损耗，满足业务对服务器的高计算性能、高稳定性、高实时性的诉求。

1.5 天翼云物理机的可用分区

1.5.1 区域和可用区的概念

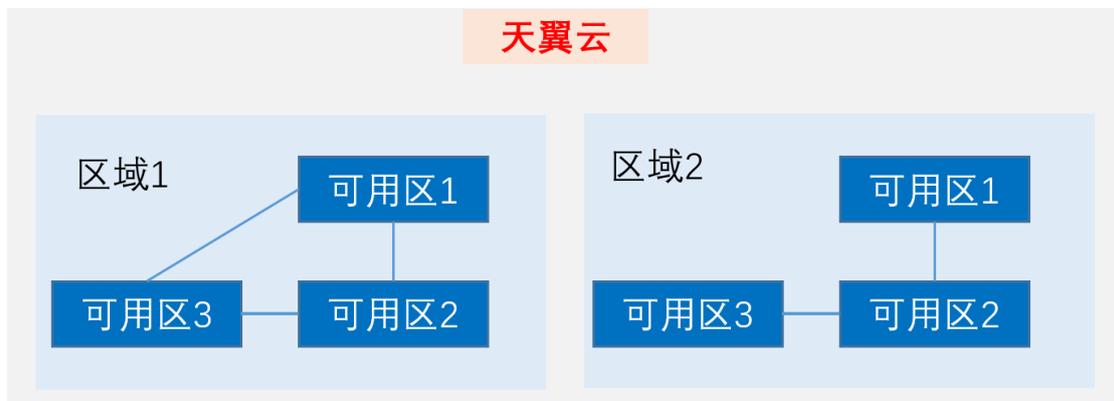
本章节主要介绍区域和可用区的概念。

我们用区域和可用区来描述数据中心的位置，您可以在特定的区域、可用区创建资源。

- 区域（Region）指物理的数据中心。每个区域完全独立，这样可以实现最大程度的容错能力和稳定性。资源创建成功后不能更换区域。
- 可用区（AZ, Availability Zone）：是同一区域内，电力和网络互相隔离的物理区域，一个可用分区不受其他可用分区故障的影响。一个地域内可以有多个可用分区，不同可用分区之间物理隔离，但内网互通，既保障了可用分区的独立性，又提供了低价、低延迟的网络连接。

若您需要确保服务的高可靠性，可以将其部署在两个可用分区。

区域和可用区之间的关系：



1.5.2 如何选择区域？

一般情况下，建议就近选择靠近您或者您的目标用户的区域，这样可以减少网络时延，提高访问速度。

1.5.3 如何选择可用区？

是否将资源放在同一可用区内，主要取决于您对容灾能力和网络时延的要求。

- 如果您的应用需要较高的容灾能力，建议您将资源部署在同一区域的不同可用区内。
- 如果您的应用要求实例之间的网络延时较低，则建议您将资源创建在同一可用区内。

1.6 天翼云物理机的相关存储类型

本章节主要介绍天翼云物理机支持挂载哪些存储类型。

物理机支持挂载云硬盘，并且支持从云硬盘启动，解决了传统物理服务器受限于本地硬盘容量的问题。

物理机还支持挂载共享云硬盘，由多台服务器并发读写访问，满足您企业核心系统集群部署的需求。

云硬盘类型：

- **普通 IO：**该类型云硬盘的最大 IOPS 为 1000，适用于大容量、读写速率中等、事务性处理较少的应用场景，例如企业的日常办公应用或者小型测试等。
- **高 IO：**该类型云硬盘的最大 IOPS 可达 3000，最低读写时延为 1ms，适用于主流的高性能、高可靠应用场景，例如企业应用、大型开发测试以及 Web 服务器日志等。
- **超高 IO：**该类型云硬盘的最大 IOPS 可达 20000，最低读写时延为 1ms，适用于超高 IO，超大带宽的读写密集型应用场景，例如高性能计算应用场景，用来部署分布式文件系统，或者 I/O 密集型应用场景，用来部署各类 NoSQL/关系型数据库。

云硬盘性能：

云硬盘性能的主要指标有 IO 读写时延、IOPS 和吞吐量。

- **IOPS：**云硬盘每秒进行读写的操作次数。
- **吞吐量：**云硬盘每秒成功传送的数据量，即读取和写入的数据量。
- **IO 读写时延：**云硬盘连续两次进行读写操作所需要的最小时间间隔。

云硬盘性能指标的详细介绍请参见《云硬盘用户指南》。

云硬盘的磁盘模式

物理机当前仅支持 SCSI（小型计算机系统接口，Small Computer System Interface）磁盘模式的云硬盘。

您可以通过管理控制台创建 SCSI 类型的云硬盘，该类型的云硬盘支持 SCSI 指令透传，允许物理机操作系统直接访问底层存储介质。除了简单的 SCSI 读写命令，SCSI 类型的云硬盘还可以支持更高级的 SCSI 命令。

说明

物理机镜像的操作系统中已经预安装了使用 SCSI 类型云硬盘所需的驱动，无需再安装。

1.7 天翼云物理机的相关网络

1.7.1 虚拟私有云

本章节主要介绍虚拟私有云。

使用虚拟私有云（Virtual Private Cloud，VPC），您可以在您自己的逻辑隔离区域中定义虚拟网络，为物理机构建一个逻辑上完全隔离的专有区域。您还可以在 VPC 中定义安全组、VPN、IP 地址段、带宽等网络特性，方便管理、配置物理机的内部网络，进行安全、快捷的网络变更。同时，您可以自定义安全组内与组间的访问规则，加强物理机的安全保护。

1.7.2 高速网络

本章节主要介绍天翼云物理机的高速网络。

物理机的内部网络，为同一可用分区内的物理机之间提供带宽不受限制的网络。如果您需要部署高吞吐量或要求低时延的服务，可以创建高速网络。目前，物理机服务支持的高速网络带宽的最大值为 10Gbps。

增强高速网络基于上一代高速网络进行了软硬件的优化升级，为租户提供更强大的网络功能。

相比上一代高速网络，增强高速网络具有如下优势：

- 带宽提升至 10GE 及以上。
- 租户自定义网络平面数量，最多支持 4K 个子网。

1.7.3 自定义 VLAN 网络

本章节主要介绍天翼云物理机的自定义 VLAN 网络。

未被系统默认使用的以太网卡（10GE，在物理机规格中定义）可用于自定义 VLAN 网络，物理上采用 QinQ 技术实现用户的网络隔离，提供额外的物理平面和网络带宽。用户能够自由划分所需的 VLAN 子网来分隔流量，适用于 Oracle RAC、SAP HANA 和 VMware 等场景。自定义 VLAN 网络的网卡是成对出现的，用户可以通过配置 bond 实现高可用。自定义 VLAN 网络当前不支持跨可用分区互通。

说明

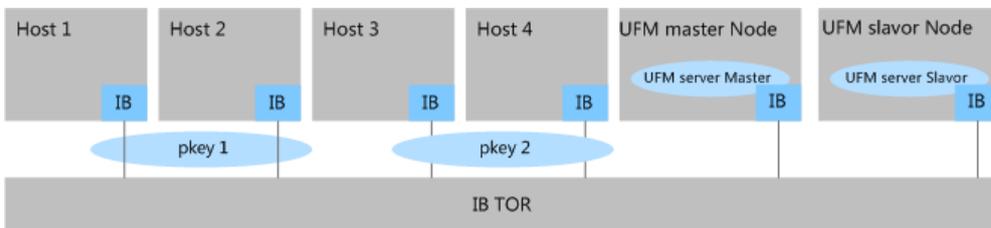
QinQ: 一种基于 802.1Q 封装的二层隧道协议，它将用户私网 VLAN TAG 封装在公网 VLAN TAG 中，报文带着两层 TAG 穿越服务商的骨干网络，从而为用户提供二层 VPN 隧道。

1.7.4 IB 网络

本章节主要介绍天翼云物理机的 IB 网络。

IB 网络因其低延迟、高带宽的网络特性被用于很多高性能计算（High Performance Computing, HPC）项目，IB 网络采用了 100G Mellanox IB 网卡，通过专用 IB 交换机和控制器软件 UFM 实现网络通信和管理。IB 网络通过 Partition Key 实现网络隔离，不同租户的 IB 网络可通过不同的 Partition Key 来隔离，类似于以太网的 vlan（如下图所示）。

图 IB 网络隔离方式：



说明

UFM（Unified Fabric Manager）是 IB 网络上 IB 交换机控制器，基于 OpenSM 软件，提供

北向服务接口。UFM 采用主备部署方式。

1.8 物理机实例规格说明

1.8.1 实例概述

物理机实例即您创建的一台物理机服务器。不同实例类型提供不同的计算能力、存储空间、网络性能，您可以基于业务需求选择不同类型的实例。当天翼云向您交付一个实例时，您将获得这台服务器完整的控制权限，包括开机、关机、带内管理等。

1.8.1.1 实例类型

目前天翼云提供的物理机 CPU，均为 x86 架构，根据业务需求选购不同配置的物理机服务器

- x86 V4 实例（CPU 采用 Intel Broadwell 架构）
- x86 V5 实例（CPU 采用 Intel Skylake 架构）
- x86 V6 实例（CPU 采用 Intel Cascade Lake 架构）

1.8.1.2 其他说明

- 基于本地盘的物理机服务器，系统盘默认 RAID 1，数据盘默认直通盘。如果需要更改数据盘 RAID 配置，可以联系管理员变更。系统盘 RAID 不支持变配。

1.8.2 x86 V4 实例（CPU 采用 Intel Broadwell 架构）

- 通用型

提供均衡的计算、存储以及网络配置，满足资源专享、网络隔离、性能有基本要求的业务场景，如数据库、核心 ERP 系统、金融系统等。

通用规格详情：

规格名称	业务场景	CPU	内存	本地磁盘	扩展配置
------	------	-----	----	------	------

physical.s2.large	数据库、大数据	20cores Intel(R) Xeon(R) CPU E5-2618L v4 (2*10core*2.20GHz)	128GB	SAS 2*600G	2*10GE
physical.s2.xlarge	数据库、大数据、核心ERP系统	28cores Intel(R) Xeon(R) CPU E5-2658 v4 (2*14core*2.30GHz)	256GB	SAS 2*600G	2*10GE
physical.s3.large	数据库、大数据	20cores Intel Xeon E5-2618 V4 (2*10core, 2.20GHz)	128GB	SAS 2*600G	2*2*10GE
physical.s3.xlarge	数据库、大数据、核心ERP系统	28cores Intel Xeon E5-2658 V4 (2*14core, 2.30GHz)	256GB	SAS 2*600G	2*2*10GE

- **本地存储型**

系统盘和数据盘均使用本地磁盘，针对数据量大，对计算性能、稳定性、实时性等要求很高的业务场景，如大数据、分布式缓存等。

规格名称	业务场景	CPU	内存	本地磁盘	扩展配置
physical.d1.large	数据库、大数据	20cores Intel(R) Xeon(R) CPU E5-2618L v4(2*10core)*2.20GHz	128GB	SATA 8*4T	4*10GE

1.8.3 x86 V5 实例 (CPU 采用 Intel Skylake 架构)

- **通用型**

提供均衡的计算、存储以及网络配置，支持挂载可弹性扩展的云硬盘，满足资源专享、网络隔离、性能有基本要求的业务场景，如数据库、核心 ERP 系统、金融系统等。

规格名称	业务场景	CPU	内存	本地磁盘	扩展配置
physical.s4.medium	数据库 核心 ERP 系统 金融 系统	20cores Intel Xeon Silver 4114 V5 (2*10core, 2.2 GHz)	128GB	NA	2x2*10GE
physical.s4.large	数据库 核心 ERP 系统 金融 系统	20cores Intel Xeon Silver 4114 V5 (2*10core, 2.20 GHz)	192GB	NA	2x2*10GE
physical.s4.xlarge	数据库 核心 ERP 系统 金融 系统	28cores Intel Xeon Gold 5120 V5 (2*14core, 2.2 GHz)	192GB	NA	2x2*10GE
physical.s4.2xlarge	高性能计算 数据库 核心 ERP 系统 金融 系统	28cores Intel Xeon Gold 5120 V5 (2*14core, 2.20 GHz)	384GB	NA	2x2*10GE
physical.s4.3xlarge	高性能计算 数据库 核心 ERP 系统 金融 系统	44cores Intel Xeon E5- 6161 (2*22core, 2.2 GHz)	384GB	NA	2x2*10GE

- **本地存储型**

系统盘和数据盘均使用本地磁盘，针对数据量大，对计算性能、稳定性、实时性等要求很高的业务场景，如大数据、分布式缓存等。

规格名称	业务场景	CPU	内存	本地磁盘	扩展配置
physical.d2.large	数据库、大数据	24cores Inter(R) Xeon(R) Gold 5188 (2*12core*2.3GHz)	192GB	2*600G SAS System Disk RAID 1 + 12*10T SATA	2 x 2*10GE

- **高性能计算型**

主要使用在受计算限制的高性能处理器的应用程序上。它需要更多处理器核数、大量的内存和高吞吐量的存储系统。该规格使用 V5 CPU 服务器，并结合 IB 网卡，适用于 HPC 高性能计算等场景。

规格名称	业务场景	CPU	内存	本地磁盘	扩展配置
physical.h2.large	数据库、大数据	36cores Intel(R) Xeon(R) Gold 2*6151(2*18core)3.00GHz	192GB	1*1.6TB NVME	2*10GE + 1*100G IB

- **GPU 加速型**

提供优秀的浮点计算能力，从容应对高实时、高并发的海量计算场景。特别适合于深度学习、科学计算、CAE、3D 动画渲染、CAD 等应用。

规格名称	业务场景	CPU	内存	本地磁盘	扩展配置
------	------	-----	----	------	------

physical.p3.large	数据库、大数据	36cores Intel(R) Xeon(R) CPU Gold 6151 (2*18core* 3.00GHz)	512GB	SAS 2*1200G + SSD 6*800G	2*10GE
-------------------	---------	--	-------	--------------------------------	--------

1.8.4 X86 V6 实例（CPU 采用 Intel Cascade Lake 架构）

- 通用型

提供均衡的计算、存储以及网络配置，支持挂载可弹性扩展的云硬盘，满足资源专享、网络隔离、性能有基本要求的业务场景，如数据库、企业 ERP 系统、容器、大数据计算等。

规格名称	业务场景	CPU	内存	本地磁盘	扩展配置
physical.c6s.xlarge	数据库、大数据、容器	52cores Intel Cascade Lake 2288X V5 (2*26 core CPU, 2.6 GHz)	192GiB	无	SDI3.0
physical.c6s.3xlarge	数据库、大数据、容器	52cores Intel Cascade Lake 2288X V5 (2*26 core CPU, 2.6 GHz)	384GiB	无	SDI3.0
physical.c6sd.3xlarge	大数据存算分离	52cores Intel Cascade Lake 2288X V5 (2*26 core CPU, 3.0 GHz)	384GiB	4*3.2T NVMe	SDI3.0 (2*25GE)

physical.s6.xlarge	数据库	24cores Intel Cascade Lake 2288X V5 (2*12 core CPU, 2.2 GHz)	192GiB	无	SDI3.0 (2*25GE)
--------------------	-----	--	--------	---	--------------------

- **高性能计算型**

采用 Intel Cascade Lake CPU，满足多核高主频、低时延 HPC 的业务场景。

规格名称	业务场景	CPU	内存	本地磁盘	扩展配置
physical.h6.xlarge	HPC	44cores Intel Cascade Lake 2288X V5 (2*22 core CPU, 3.0 GHz)	192GiB	无	SDI3.0 (2*25GE)

- **GPU 加速型**

采用 Intel Cascade Lake CPU、NVIDIA T4，满足 AI 推理和图形图像加速业务场景。

规格名称	业务场景	CPU	内存	本地磁盘	扩展配置
physical.pi6.3xlarge.6	异构图形图像，AI 推理	52cores ATLAS G560V5 (2*26 core CPU, 2.6 GHz)	448GiB	2* 480G SATA SSD+1*1600G NVMe SSD	6*T4 + 2*Hi1822

1.9 天翼云物理机的镜像

1.9.1 镜像概述

镜像是一个至少包含操作系统，还可以包含应用软件（例如，数据库软件）及软件必要配置的物理机模板。

镜像分为公共镜像和私有镜像，公共镜像为系统默认提供的镜像，私有镜像为用户自己创建的镜像，用户也可以提工单给天翼云运维团队协助您完成私有镜像的创建。您可以灵活便捷地使用公共镜像或者私有镜像申请物理机。同时，用户还能通过已有的物理机或者外部镜像文件创建私有镜像，这样可以快速轻松地创建能满足您一切需求的物理机。

镜像类型

根据镜像来源可以划分为公共镜像、私有镜像和共享镜像。

镜像类型	说明
公共镜像	常见的标准操作系统镜像，所有用户可见，包括操作系统以及预装的公共应用或服务。
私有镜像	包含操作系统、预装的公共应用以及用户的私有应用，仅用户个人可见。选择私有镜像创建物理机服务器，可以节省您重复配置物理机服务器的时间。（用户可以提工单，让运维后台协助您快速完成私有镜像的制作。）
共享镜像	由其他用户共享的私有镜像。

1.9.2 物理机服务器类型与支持的操作系统版本

物理机服务器不同规格操作系统兼容性列表如下所示。操作系统版本均为 64bit。

物理机服务器兼容性列表（V4）

类型	Windows	CentOS	Red Hat	SUSE	Ubuntu	Oracle Linux
本地存储型 d1 physical.d1.large	Windows Server 2012 R2 Standard	CentOS 6.8/7.4/7.5/7.6	Red Hat 6.8	-	Ubuntu 16.04 LTS	-
通用型规格 s3 physical.s3.large physical.s3.xlarge physical.s3.2xlarge	Windows Server 2012 R2 Standard	CentOS 7.3/7.4/7.5	Red Hat 6.8	SUSE Linux Enterprise 11.SP4/12.SP1/ 12.SP2/12.SP3	Ubuntu 16.04 LTS	Oracle Linux 6.9/7.3
GPU 加速型规格 physical.p1.large physical.p2.large	-	CentOS 7.4	-	-	Ubuntu 16.04 LTS	-

物理机服务器兼容性列表（V5）

类型	Windows	CentOS	Red Hat	SUSE	Ubuntu	Oracle Linux
本地存储型 d2 physical.d2.large	Windows Server 2012 R2 Standard	CentOS 6.8/7.2/7.3/7.4/ 7.5	Red Hat 6.7/6.8	SUSE Linux Enterprise 11.SP4/12.SP1 /12.SP2/12.SP 3	Ubuntu 16.04 LTS	Oracle Linux 6.9/7.3

通用型规格 s4 physical.s4.medium physical.s4.large physical.s4.xlarge physical.s4.2xlarge physical.s4.3xlarge	Windows Server 2012 R2 Standard	CentOS 7.4/7.5	Red Hat 6.7/6.8	SUSE Linux Enterprise 11.SP4	Ubuntu 16.04 LTS	Oracle Linux 6.9/7.3
IO 优化型规格 io2 physical.io2.xlarge	Windows Server 2012 R2 Standard	CentOS 6.9/7.3/7.4/7.5	Red Hat 6.9/7.3/7.4/7.5	SUSE Linux Enterprise 11.SP4	Ubuntu 16.04 LTS	Oracle Linux 6.9/7.4
高性能计算型规格 h2 physical.h2.large	-	CentOS 6.9/7.3/7.4	-	SUSE Linux Enterprise 11.SP4	Ubuntu 16.04 LTS	-
GPU 加速型规格 p3 physical.p3.large	-	CentOS 7.3	-	-	-	-

物理机服务器兼容性列表 (V6)

类型	CentOS	Red Hat	SUSE	Ubuntu	Oracle Linux
通用型规格 s6/c6 physical.s6.xlarge physical.c6s.xlarge physical.c6s.3xlarge	CentOS 7.6/8.2	Red Hat 7.6	SUSE Linux Enterprise 15.SP2	Ubuntu 18.04.3 LTS	Oracle Linux7.6

e physical.c6sd.3xlarge					
高性能计算型规格 h6 physical.h6.large	CentOS 7.6	-	-	-	-
本地存储型规格 d6 physical.d6.xlarge	CentOS 7.6	Red Hat 7.6	SUSE Linux Enterprise 15.SP2	Ubuntu 18.04.3 LTS	Oracle Linux7.6
GPU 加速型规格 pi6 physical.pi6.3xlarge e.6	CentOS 7.6	-	-	-	-

1.10 天翼云物理机的安全

1.10.1 安全组

1.10.1.1 安全组介绍

安全组是一种虚拟防火墙，具备状态检测和数据包过滤功能，用于设置弹性云服务器、物理机服务器、负载均衡、数据库等实例的网络访问控制，是重要的网络安全隔离手段。

您可以通过配置安全组规则，允许安全组内的实例对公网或私网的访问。

- 安全组是一个逻辑上的分组，您可以将同一区域内具有相同安全保护需求的物理机服务器加入到同一个安全组内。
- 同一安全组内的 **BMS** 实例之间默认内网网络互通，不同安全组内的实例之间默认内网不通。
- 您可以随时修改安全组的规则，新规则立即生效。

1.10.2 密钥对与密码

1.10.2.1 密钥对概述

密钥对，也称 SSH 密钥对，是区别于用户名+密码的远程登录 Linux 实例的认证方式。通过加密算法生成一对密钥，一个对外界公开，称为公钥，另一个由用户自己保留，称为私钥。公钥和私钥组成密钥对。密钥对的工作原理是使用公钥加密某个数据（例如一个密码），用户可以使用私钥解密数据。

天翼云只会存储公钥，您需要存储私钥。拥有您的私钥的任何人都可以解密您的登录信息，因此将您的私钥保存在一个安全的位置非常重要。

1.10.2.2 功能优势

相比用户名+密码认证方式，密钥对在安全性和便捷性上都更具优势，如表 1 所示。

表 1 密钥对与密码对比

对比项	密钥对	用户名 + 密码
安全性	安全强度远高于常规用户口令，可以杜绝暴力破解威胁。 不可能通过公钥推导出私钥。	安全性较低。
便捷性	便于远程登录大量 Linux 实例，方便管理。	一次只能登录一台 Linux 实例，不利于批量维护。

1.10.2.3 使用限制

本章节主要介绍天翼云物理机密钥对的使用限制。

- 仅支持 Linux 实例。
- 仅支持的 RSA 密钥对，支持的长度：1024、2048 和 4096。
- 一台 Linux 实例只能拥有一个密钥对。如果您的实例已绑定密钥，绑定新的密钥会替换原来的密钥。

1.10.2.4 生成方式

本章节主要介绍天翼云物理机密钥对的生成方式。

- 通过管理控制台创建密钥对

说明

在首次生成密钥对时，务必下载并妥善保存私钥。

- 通过 PuTTYgen 等工具创建密钥对，并导入天翼云

具体生成方式可参考操作指南中的[密钥与密码](#)介绍。

1.11 物理机的访问和使用

本章节主要介绍天翼云物理机的访问和使用方式。

访问物理机

天翼云控制平台提供了 Web 化的服务管理系统（即管理控制台）的管理方式

管理控制台方式

其他相关操作，请使用管理控制台方式访问物理机。如果用户已注册天翼云，可直接登录管理控制台，从主页选择“物理机”。

如果未注册，请注册天翼云。

远程登录登录方式

如果物理机配置有弹性 IP，则 Windows 物理机使用远程登录方式登录，Linux 物理机使用 ssh 方式登录。

1.12 物理机权限管理

本章节主要介绍物理机权限管理。

如果您需要对天翼云上购买创建的物理机服务器资源，为企业中的员工设置不同的访问权限，以达到不同员工之间的权限隔离，您可以使用统一身份认证服务（Identity and Access

Management, 简称 IAM) 进行精细的权限管理。该服务提供用户身份认证、权限分配、访问控制等功能, 可以帮助您安全的控制天翼云云服务资源的访问。

通过 IAM, 您可以在帐号中给员工创建 IAM 用户, 并授权控制他们对天翼云资源的访问范围。例如您的员工中有负责软件开发的人员, 您希望他们拥有物理机的使用权限, 但是不希望他们拥有删除物理机等高危操作的权限, 那么您可以使用 IAM 为开发人员创建用户, 通过授予仅能使用物理机, 但是不允许删除物理机的权限, 控制他们对物理机资源的使用范围。

如果天翼云帐号已经能满足您的要求, 不需要创建独立的 IAM 用户进行权限管理, 您可以跳过本章节, 不影响您使用物理机服务的其它功能。

IAM 是天翼云提供权限管理的基础服务, 无需付费即可使用, 您只需要为您帐号中的资源进行付费。

物理机权限

默认情况下, 管理员创建的 IAM 用户没有任何权限, 需要将其加入用户组, 并给用户组授予策略或角色, 才能使得用户组中的用户获得对应的权限, 这一过程称为授权。授权后, 用户就可以基于被授予的权限对云服务进行操作。

物理机部署时通过物理区域划分, 为项目级服务。授权时, “作用范围” 需要选择“区域级项目”, 然后在指定区域(如江苏-苏州)对应的项目(cn-jssz1)中设置相关权限, 并且该权限仅对此项目生效; 如果在“所有项目”中设置权限, 则该权限在所有区域项目中都生效。

访问物理机时, 需要先切换至授权区域。

根据授权精度分为角色和策略。

- **角色:** IAM 最初提供的一种根据用户的工作职能定义权限的粗粒度授权机制。该机制以服务为粒度, 提供有限的服务相关角色用于授权。由于天翼云云服务平台各服务之间存在业务依赖关系, 因此给用户授予角色时, 可能需要一并授予依赖的其他角色, 才能正确完成业务。角色并不能满足用户对精细化授权的要求, 无法完全达到企业对权限最小化的安全管控要求。
- **策略:** IAM 最新提供的一种细粒度授权的能力, 可以精确到具体服务的操作、资源以及请求条件等。基于策略的授权是一种更加灵活的授权方式, 能够满足企业对权限最小化的安全管控要求。例如: 针对 BMS 服务, 管理员能够控制 IAM 用户仅能对某一类裸金属服务器资源进行指定的管理操作。多数细粒度策略以 API 接口为粒度进行权限拆分。

物理机系统权限如下表:

系统角色/策略名称	描述	类别
-----------	----	----

BMS FullAccess	裸金属服务器管理员，拥有该服务下的所有权限	系统策略
BMS CommonOperations	裸金属服务器基本操作权限，包括开机、关机、重启、查询等操作	系统策略
BMS ReadOnlyAccess	裸金属服务器只读权限，拥有该权限的用户仅能查看裸金属服务器列表和详情	系统策略

物理机常用操作与系统权限的关系 如下表：

操作	BMS FullAccess	BMS CommonOperations	BMS ReadOnlyAccess
创建裸金属服务器	√	x	x
查看裸金属服务器列表	√	√	√
查询裸金属服务器详情	√	√	√
修改裸金属服务器名称	√	x	x
启动裸金属服务器	√	√	x
关闭裸金属服务器	√	√	x
重启裸金属服务器	√	√	x
裸金属服务器挂载数据卷	√	√	x
裸金属服务器卸载数据卷	√	√	x
重装裸金属服务器操作系统	√	x	x
一键重置裸金属服务器的密码	√	x	x

1.13 物理机与其他服务的关系

云硬盘服务

可以将云硬盘挂载至物理机，并可以随时扩容云硬盘容量。

虚拟私有云

为物理机提供一个逻辑上完全隔离的专有网络，您还可以在 VPC 中定义安全组、VPN、IP 地址段、带宽等网络特性。用户可以通过 VPC 方便地管理、配置内部网络，进行安全、快捷的网络变更。同时，用户可以自定义安全组内与组间物理机的访问规则，加强物理机的安全保

护。

镜像服务

您可以通过镜像创建物理机，提高物理机的部署效率。

云监控服务

当您开通了物理机后，安装 `telescope` 软件并完成相关配置，即可在云监控服务查看物理机的监控数据，还可以获取可视化监控图表。如何配置物理机带内监控，请参考[《主机监控配置》](#)。

1.14 天翼云物理机的使用限制

本章节主要介绍天翼云物理机的使用限制。

实例

- 支持故障重建：由于服务器硬件损坏、SDI 卡损坏等原因，导致服务器无法正常使用时，可以申请服务器重建。【无本地盘机型支持】【运维发起】。
- 支持用户自主管理物理机服务器生命周期：查询、启动、关机、重启、删除。
- 支持操作系统一键重置密码。
- 支持导出服务器列表：将租户名下的所有物理机服务器信息，以 CSV 文件的形式导出至本地。
- 支持云审计，记录与物理机相关的操作事件，便于日后的查询、审计和回溯。
- 支持主机监控，可实时获取物理机服务器的 CPU、内存、磁盘 I/O 等监控指标数据。
- 支持标签管理服务，使用标签来标识物理机，便于分类和搜索。

磁盘

- 支持挂载/卸载云硬盘（linux 和 windows 均支持）。
- 支持挂载共享卷。
- 支持磁盘动态扩容。

镜像

- 可以使用公共镜像、私有镜像、共享镜像发放物理机。
- 支持通过物理机服务器创建私有镜像。
- 支持通过外部镜像文件创建私有镜像。
- 私有镜像还支持不同帐号之间共享镜像。

网络

- 支持虚拟私有云、自定义 VLAN 网络、高速网络、IB 网络，满足多种场景的需求。
- 创建安全组并定义安全组规则，为物理机提供安全防护。
- 支持绑定弹性公网 IP，满足客户互联网访问需求。
- 支持绑定多张网卡，支持动态绑定网卡。

使用限制

- 不支持直接加载外接硬件设备（如 USB 设备、银行 U key、外接硬盘、加密狗等）。
- 不支持带外管理，您的物理机资源统一由天翼云管理和维护。
- 不支持创建没有操作系统的裸设备，即物理机必须自带操作系统。
- 不支持热迁移，服务器故障后会对业务造成影响，建议您通过业务集群部署、主备部署等方式实现业务的高可用。
- 不支持更换物理机的操作系统。
- 创建后不支持更换 VPC。
- 公共镜像的 Windows 物理机系统是默认激活的，您在上面二次虚拟化出来的操作系统，需要您自行激活。
- 不支持自定义物理机配置，不支持规格变配，仅支持云硬盘在线扩容。
- 仅支持挂载 SCSI 类型的云硬盘。
- 由于某些机型的服务器没有配备 SDI 卡，或者其他服务器本身的原因，有些规格或镜像的物理机不支持挂载云硬盘。
- 请勿删除或者修改镜像中内置的插件服务（如 Cloud-Init、bms-network-config 等），否则会影响您的基本功能使用。

- 在创建物理机时，如果选择自动分配 IP 地址，请不要在物理机发放完成后修改私有 IP 地址，避免和其他物理机 IP 冲突。
- 物理机不支持配置桥接网卡，会导致网络不通。
- 禁止升级 OS 自带内核版本，否则服务器硬件驱动会存在兼容性风险，影响服务器可靠性。
- 物理机禁止修改和配置系统盘。

2 购买指南

2.1 物理机规格和价格

本章节主要介绍天翼云物理机的规格和价格。

V4 规格和价格

规格名称	业务场景	CPU	内存	本地磁盘	扩展配置	价格 (元/月)
physical.d1.large	数据库、大数据	20cores Intel (R)	128GB	SATA 8*4T	4*10GE	5984

		Xeon (R) CPU E5-2618L v4(2*10core) *2. 20GHz				
physical. s3. large	数据库、大数 据	20cores Intel Xeon E5-2618 V4 (2*10core, 2. 20GHz)	128GB	SAS 2*600G	2*2*10GE	5328
physical. s3. xlarge	数据库、大数 据、核心 ERP 系统	28cores Intel Xeon E5-2658 V4 (2*14core, 2. 30GHz)	256GB	SAS 2*600G	2*2*10GE	7824

V5 规格和价格

规格名称	业务场景	CPU	内存	本地磁盘	扩展配置	价格 (元/ 月)
physical. s4. medium	数据库 核心 ERP 系统 金 融系统	20cores Intel Xeon Silver 4114 V5 (2*10core, 2. 2 GHz)	128GB	NA	2x2*10GE	5280
physical. s4. large	数据库 核心 ERP 系统 金 融系统	20cores Intel Xeon Silver 4114 V5	192GB	NA	2x2*10GE	5600

		(2*10core, 2.20 GHz)				
physical.s4. xlarge	数据库 核心 ERP 系统 金 融系统	28cores Intel Xeon Gold 5120 V5 (2*14core, 2.2 GHz)	192GB	NA	2x2*10GE	7456
physical.s4. 2xlarge	高性能计算 数据库 核心 ERP 系统 金 融系统	28cores Intel Xeon Gold 5120 V5 (2*14core, 2.20 GHz)	384GB	NA	2x2*10GE	8416
physical.s4. 3xlarge	高性能计算 数据库 核心 ERP 系统 金 融系统	44cores Intel Xeon E5-6161 (2*22core, 2.2 GHz)	384GB	NA	2x2*10GE	12128

V6 规格和价格

- V6 + SDI3.0 物理机实例

规格名称	业务场景	CPU	内存	本地磁盘	扩展配置	价格 (元/月)
physical.c6s .xlarge	数据库、大数 据、容器	52cores Intel Cascade Lake 2288X V5 (2*26 core	192GiB	NA	SDI3.0	13024

		CPU, 2.6 GHz)				
physical.c6s.3xlarge	数据库、大数据、容器	52cores Intel Cascade Lake 2288X V5 (2*26 core CPU, 2.6 GHz)	384GiB	NA	SDI3.0	13984
physical.c6sd.3xlarge	大数据存算分离	52cores Intel Cascade Lake 2288X V5 (2*26 core CPU, 3.0 GHz)	384GiB	4*3.2T NVMe	SDI3.0 (2*25GE)	17184
physical.pi6.3xlarge.6	异构图形图像, AI 推理	52cores ATLAS G560V5 (2*26 core CPU, 2.6 GHz)	448GiB	2* 480G SATA SSD+1*1600G NVMe SSD	6*T4 + 2*Hi1822	31939
physical.s6.xlarge	数据库	24cores Intel Cascade Lake 2288X V5 (2*12 core CPU, 2.2 GHz)	192GiB	NA	SDI3.0 (2*25GE)	6528
physical.s6.3xlarge	数据库	24cores Intel Cascade Lake	384GiB	NA	SDI3.0 (2*25GE)	8073.5

		2288X V5 (2*12 core CPU, 2.2 GHz)				
physical. d6. xlarge	大数据	24cores Intel Cascade Lake 2288X V5 (2*12 core CPU, 2.2 GHz)	192GiB	12*10TB SATA HDD	SDI3.0 (2*25GE)	10994
physical. d6. 3xlarge	大数据	24cores Intel Cascade Lake 2288X V5 (2*12 core CPU, 2.2 GHz)	384GiB	12*10TB SATA HDD	SDI3.0 (2*25GE)	12873.5
physical. h6. xlarge	HPC	44cores Intel Cascade Lake 2288X V5 (2*22 core CPU, 3.0 GHz)	192GiB	NA	SDI3.0 (2*25GE)	12750

• **ECS 物理机实例**

规格名称	业务场景	CPU	内存	本地磁盘	扩展配置	价格 (元/月)
------	------	-----	----	------	------	-------------

c6.22xlarge. 2.physical	数据库、大数据、容器	44cores Intel Cascade Lake 2288X V5 (2*22 core CPU, 3.0 GHz)	192GiB	NA	SDI3.0 (2*25GE)	11168
c6.22xlarge. 4.physical	数据库、大数据、容器	44cores Intel Cascade Lake 2288X V5 (2*22 core CPU, 3.0 GHz)	384GiB	NA	SDI3.0 (2*25GE)	12128
m6.22xlarge. 8.physical	大数据存算分离	44cores Intel Cascade Lake 2288X V5 (2*22 core CPU, 3.0 GHz)	768GiB	NA	SDI3.0 (2*25GE)	14048

2.2 物理机计费项

本章节主要介绍天翼云物理机的计费项说明。

计费项

计费项	计费说明
物理机	对您选择的 BMS 实例规格计费（包括 CPU 规格、内存容量、本地硬盘和扩展配置等）。
云硬盘（可选）	随物理机一起创建的云硬盘，计费方式与物理机相同。单独创建的云硬盘，支持按量付费和包年包月。
弹性 IP（可选）	物理机实例支持绑定弹性 IP，按带宽计费。

计费模式：

天翼云物理机计费模式为包年包月付费，不支持按量付费。对于长期使用者，推荐包年购买，有优惠。

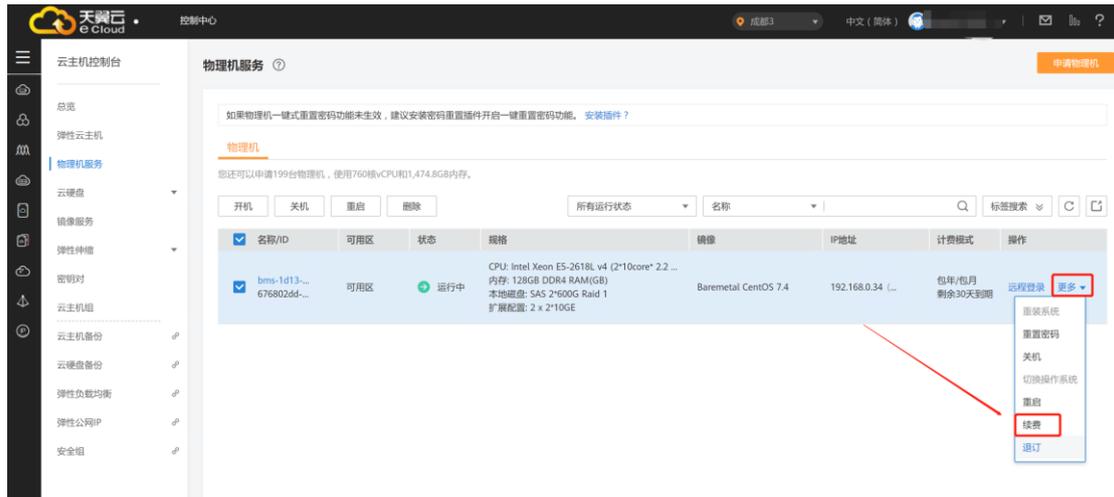
变更配置

扩容存储空间：如果您的业务需求增大，可以通过扩容或者增加物理机挂载的云硬盘来扩展您的存储空间。这部分存储空间将按照您选择的云硬盘计费方式（按需或包周期）计费。

续订规则

续订物理机，续费可按月或按年进行续费，购买后用户可自定义续费周期。

如果需要为某一物理机续费，请单击“操作”列下的“更多 > 续费”，跳转至续费管理页面完成续费。



自动续订：用户购买物理机后，在管理中心 → 产品中心 → 产品续订，选中物理机产品可开通自动续订。具体续订操作说明，请参考费用中心 → 续订 → [自动续订说明](#)。

退订规则

退订云物理机后实例会进入冻结状态，冻结周期为 15 天，冻结期过后如不续费会删除实例。具体退订规则说明，请参考费用中心--[退订规则说明](#)。

提醒/通知规则

- 到期通知：服务到期前 7 天、3 天邮件通知，到期前 1 天、当天邮件通知和短信提醒。
- 超期通知：服务超期 1 天邮件通知，超期 3 天、7 天邮件通知和短信提醒。

到期与欠费

• 到期

到期是针对“包年/包月”的产品而言的，“按需计费”的产品，没有到期的概念。

“包年/包月”实例到期后无法在物理机管理控制台进行该实例的操作，相关接口也无法调用，自动化监控或告警等运维也会停止。如果在冻结期结束时您没有续费，实例将终止服务，系统中的数据也将被永久删除。

• 欠费

欠费是针对“按需计费”的产品和账户而言的，“包年/包月”的产品，没有欠费的概念。

“按需计费”实例是按每小时扣费，当余额不足，无法对上一个小时的费用进行扣费，就会导致实例欠费。您续费后解冻实例，可继续正常使用。

各使用周期内的资源状态

使用周期	状态	计费模式
正常使用周期内	运行中	包年/包月 剩余 xx 天到期
超期未续费，进入冻结期	冻结	包年/包月 已冻结，xx 天后删除
超期未续费，已过冻结期	已删除	包年/包月 删除

冻结、解冻、释放资源时对业务的影响

- 资源冻结时：资源将被限制访问和使用，会导致您的业务中断。例如物理机退订后自动进入冻结期，物理机资源冻结时，会自动关机。
- 资源解冻时：资源将被解除限制，但是需要您自行检查并恢复业务。例如物理机续费后资源自动解冻，但需要您自行开机。
- 资源释放时：物理机资源冻结期过后资源将释放被删除，存储在资源中的数据将被删除，数据无法找回。

包年/包月资源的使用建议

如果包年/包月物理机服务器到期后，您不再继续使用该服务器，则可以登录管理控制台退订删除资源。

对于以下常见的包年/包月物理机服务器的使用场景，您可以参考我们提供的建议开通自动续费功能，并关注资源到期、冻结等相关通知信息，以便及时了解资源情况，确保您的业务和数据不受影响。

2.3 物理机购买

本章节主要介绍天翼云物理机的购买说明。

自助开通

1. 登录控制中心。
2. 在服务列表页面，单击“计算 >物理机服务”。
3. 在“物理机”界面，单击“申请物理机”。
4. 选择“区域”。

不同区域的云服务产品之间内网互不相通。建议您选择最靠近您业务的区域，这样可以减少网络时延、提高访问速度。

5. 选择“可用区”。

可用区指在同一区域下，电力、网络隔离的物理区域，可用区之间内网互通，不同可用区之间物理隔离。

- 如果您需要提高应用的高可用性，建议您将物理机创建在不同的可用区。
- 如果您需要较低的网络时延，建议您将物理机创建在相同的可用区。

6. 选择“规格”。

包括规格名称、CPU、内存、本地磁盘和扩展配置。当您选择规格列表中的一个规格后，列表下方会展示该规格的名称、组网、用途等信息，以便您根据业务场景进行选择。

其中，扩展配置描述了所选机型的网卡信息。例如：**2 x 2*10GE** 表示 1 块双网口的 10GE 网卡接入 VPC 网络，1 块双网口的 10GE 扩展网卡支持物理机间的高速互联。

规格中的 CPU、内存、本地磁盘等配置为固定值，不可更改。

不同规格的物理机带宽能力不同，请您根据实际业务慎重选择。

某些规格的物理机支持快速发放能力，选择后，界面会提供“系统盘”参数（在“磁盘”配置项中展示），物理机操作系统直接安装在云硬盘，实现快速发放。

7. 选择“镜像”。

- 公共镜像

常见的标准操作系统镜像，所有用户可见，包括操作系统以及预装的公共应用（SDI 卡驱动、bms-network-config 网络配置程序、cloud-init 初始化工具等）。请根据您的实际需要自助配置应用环境或相关软件。

- 私有镜像

用户基于外部镜像文件或物理机创建的个人镜像，仅用户自己可见。包含操作系统、预装的公共应用以及用户的私有应用。选择私有镜像创建物理机，可以节省您重复配置物理机的时间。

- 共享镜像

您将接受公有云其他用户共享的私有镜像，作为自己的镜像进行使用。

8. 选择“许可类型”。

在云平台上使用操作系统或软件的许可证类型。如果您选择的镜像为免费的，则系统不会展示该参数。如果您选择的镜像为计费镜像，此时系统会展示该参数。

- 使用平台许可证

使用云平台提供的许可证，申请许可证需要支付一定的费用。

- 使用自带许可证（BYOL）

使用用户已有操作系统的许可证，无需重新申请。

9. 设置“磁盘”。

有 SDI 卡的物理机可以使用云硬盘，

磁盘包括系统盘和数据盘。您可以为物理机添加多块数据盘，系统盘大小可以根据需要自定义。

- 系统盘

如果选择支持快速发放的规格，界面会提供系统盘配置项，可以根据需要设置磁盘类型和大小。

- 数据盘

您可以为物理机添加多块数据盘，并设置数据盘的共享功能。

注意：

物理机仅支持 **SCSI** 磁盘模式。

共享盘：勾选后，数据盘为共享云硬盘。该共享盘可以同时挂载给多台物理机使用。

10. 设置“网络”，包括“虚拟私有云”、“安全组”、“网卡”等信息。

第一次使用云服务时，系统将自动为您创建一个默认的虚拟私有云，包括安全组、网卡。其中，默认虚拟私有云支持的地址范围为 **192.168.1.0/24**，子网网关为 **192.168.1.1**，并为子网开启 **DHCP** 功能。

参数	解释
虚拟私有云	您可以选择使用已有的虚拟私有云，或者单击“查看虚拟私有云”创建新的虚拟私有云。
安全组	<p>安全组用来实现安全组内和安全组间物理机的访问控制，加强物理机的安全保护。用户可以在安全组中定义各种访问规则，当物理机加入该安全组后，即受到这些访问规则的保护。</p> <p>创建物理机时，仅支持选择一个安全组。但是物理机创建成功后，可以为物理机关联多个安全组，配置方法请参见 5.2 配置安全组。</p> <p>说明</p> <p>物理机初始化需要确保安全组出方向规则至少满足如下要求：</p> <ul style="list-style-type: none"> 协议：TCP 端口范围：80 远端地址：169.254.0.0/16

参数	解释
	如果您使用的是默认安全组出方向规则，则已经包括了如上要求，可以正常初始化。默认安全组出方向规则为： <ul style="list-style-type: none"> 协议：Any 端口范围：Any 远端地址：0.0.0.0/16
网卡	包括主网卡和扩展网卡。您可以添加多张扩展网卡，并指定网卡（包括主网卡）的 IP 地址。 注意 <ul style="list-style-type: none"> 主网卡用于系统的默认路由，不允许删除。 如果您选择自动分配 IP 地址，请不要在物理机发放完成后修改私有 IP 地址，避免和其他物理机 IP 冲突。 为网卡分配固定 IP 地址后，不能批量创建物理机。
高速网卡	物理机之间的高速互连网络，为物理机提供更高的带宽。 一台物理机最多有两块高速网卡，并且依赖于扩展网卡总带宽（可以在规格的“扩展配置”列查看）。例如，扩展网卡总带宽为 2 * 10GE，如果第一块高速网卡的带宽为 2 * 10GE，那么您不能再添加第二块高速网卡。 说明 同一台物理机的多张高速网卡不能选择同一个高速网络。
弹性公网 IP	弹性公网 IP 是指将公网 IP 地址和路由网络中关联的物理机绑定，以实现虚拟私有云内的物理机通过固定的公网 IP 地址对外提供访问服务。 您可以根据实际情况选择以下三种方式： <ul style="list-style-type: none"> 不使用：物理机不能与外部网络互通，仅可以在私有网络中部署业务或构建集群。 自动分配：自动为物理机分配独享带宽的弹性公网 IP，带宽值由您设定。 使用已有：为物理机分配已有弹性公网 IP。 说明 选择已有弹性公网 IP 后，不能批量创建物理机。
规格	电信
计费方式	弹性公网 IP 选择“自动分配”时，需配置该参数。

参数	解释
	按带宽计费：指定带宽上限，按使用时间计费，与使用的流量无关。 按流量计费：按照实际使用的流量来计费。
带宽	弹性公网 IP 选择“自动分配”时，需配置该参数。 带宽大小，单位 Mbit/s。

11. 设置物理机登录方式。

“密钥对”方式创建的物理机安全性更高，建议选择“密钥对”方式。如果您习惯使用“密码”方式，请增强密码的复杂度，如下表所示，保证密码符合要求，防止被恶意攻击。

- 密钥对

指使用密钥对作为登录物理机的鉴权方式。您可以选择使用已有的密钥，或者单击“查看密钥对”创建新的密钥。

如果选择使用已有的密钥，请确保您已在本地获取该文件，否则，将影响您正常登录物理机。

- 密码

指使用设置初始密码方式作为物理机的鉴权方式，此时，您可以通过用户名密码方式登录物理机。Linux 操作系统时为 root 用户的初始密码，Windows 操作系统时为 Administrator 用户的初始密码。密码复杂度需满足要求。

注意：

Windows 操作系统不支持选择密码登录方式。

参数	规则	样例
密码	密码长度范围为 8 到 26 位。 密码至少包含以下 4 种字符中的 3 种： 大写字母	Test12\$@

参数	规则	样例
	小写字母 数字 特殊字符，包括!@\$%^-_=+[]{};./? 密码不能包含用户名或用户名的逆序。 Windows 系统的物理机，不能包含用户名中超过两个连续字符的部分。	

12. 设置“物理机名称”。

名称可自定义，但需符合命名规则：只能由中文字符、英文字母、数字及“_”、“-”、“.”组成。

一次创建多台物理机，系统会自动按序增加后缀。例如：输入 **bms**，服务器名称为 **bms-0001**，**bms-0002**，……。再次购买多台服务器时，命名从上次最大值连续增加，例如：输入 **bms**，已有服务器 **bms-0010**，新创服务器名称为 **bms-0011**、**bms-0012**、……，命名达到 9999 时，从 0001 开始。

设置您创建的物理机数量，最多为 24 台。

设置完成后，您可以通过单击“价格计算器”，查询当前配置的费用。

说明：

- 当配额充足时，一次最多可以创建 24 台物理机；若配额不足 24 台，一次最多可创建数量为配额余量。
- 在设置网络信息时，若您选择自定义网卡或高速网卡的 IP 地址，或者选择已有弹性公网 IP，则一次只能创建一台物理机。

13. 单击“立即申请”。

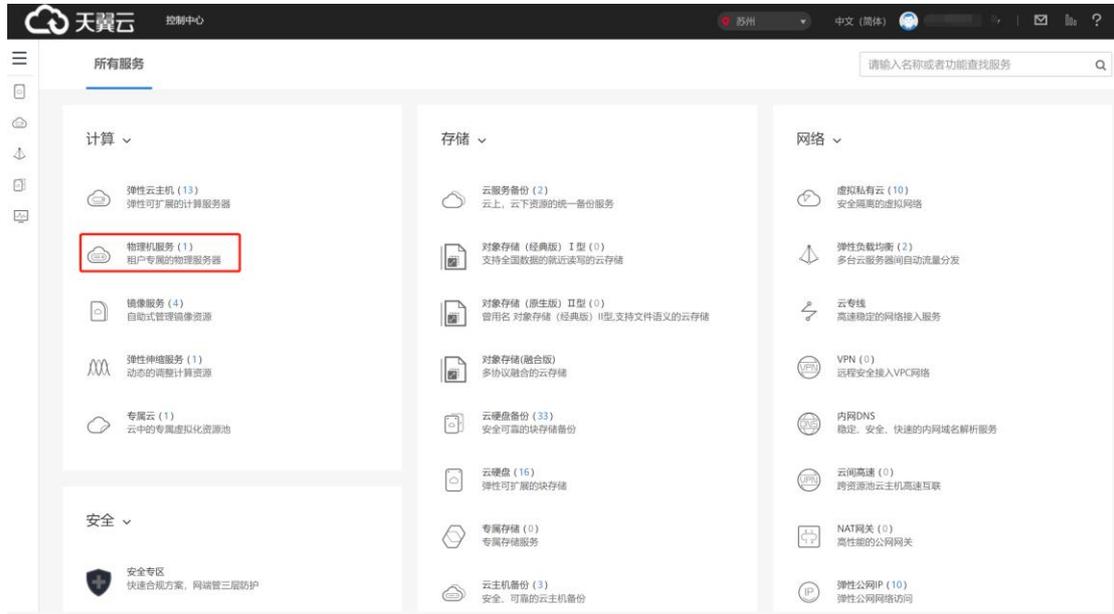
14. 在确认规格页面，单击“提交”。

注：需要拥有物理机自助开通权限，如没有权限可以立工单开通

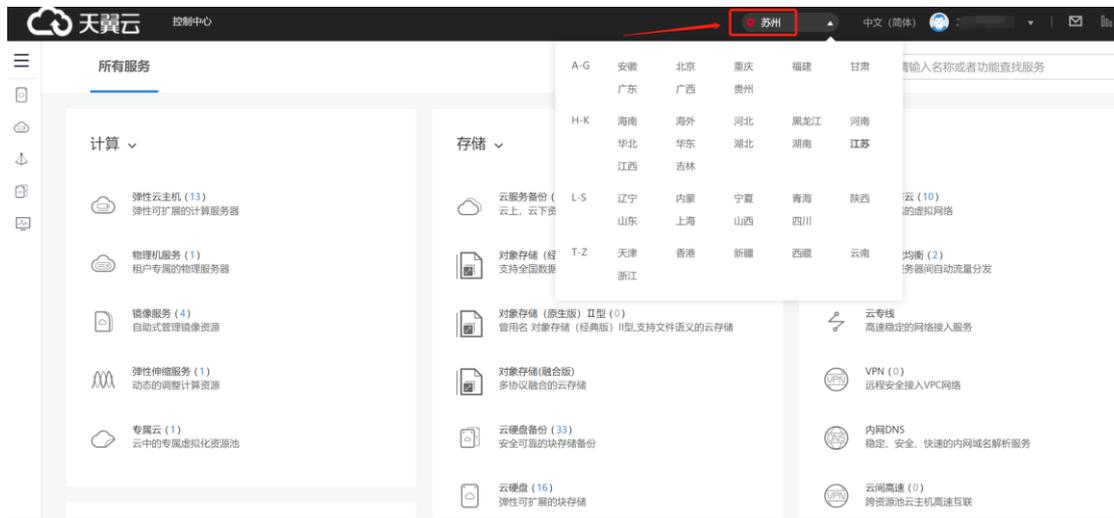
通过客户经理开通

用户开通项目制物理机请通过客户经理联系业务受理开通。

1. 用户登陆控制中心后，单击“计算 > 物理机服务”。



2. 单击管理控制中心的区域按钮，选择地域和项目。



3. 选择“计算 > 物理机服务”，进入物理机页面。如下图所示，页面提示通过拨打 400 电话联系客服开通物理机服务，联系客服，或可直接联系客户所对应的客户经理开通物理机服务。



4. 客户将具体的物理机需求（如规格、数量、网络配置信息等）告知客户经理，由客户经理完成后续的开通流程。

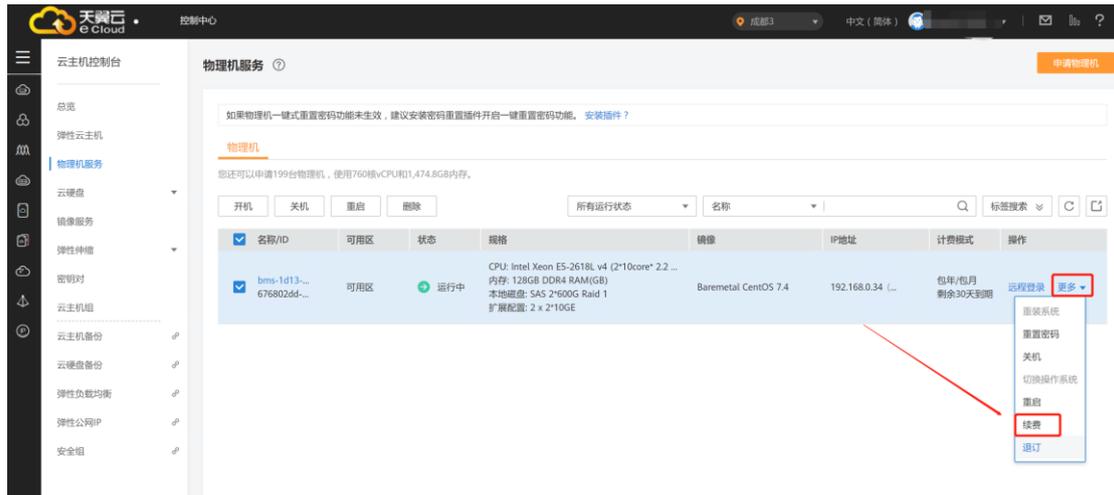
5. 物理机创建需 1-2 个工作日，创建完成后，客户将收到客户经理的通知，并可登录平台进行使用。

2.4 物理机续订

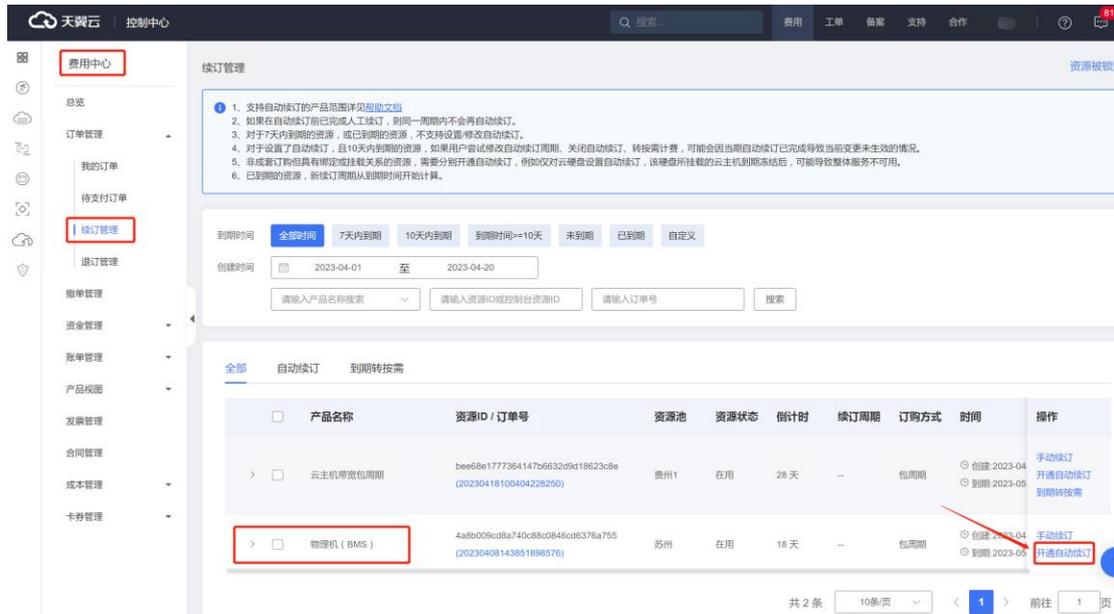
本章节主要介绍天翼云物理机的续订说明。

1. 登录管理控制台。
2. 单击管理控制台的区域按钮，选择地域和项目。
3. 选择“计算 > 物理机服务”，进入物理机页面。
4. 在物理机的“操作”列下，单击“更多 > 续费”，进入“续订”界面。
5. 点击“确定”。

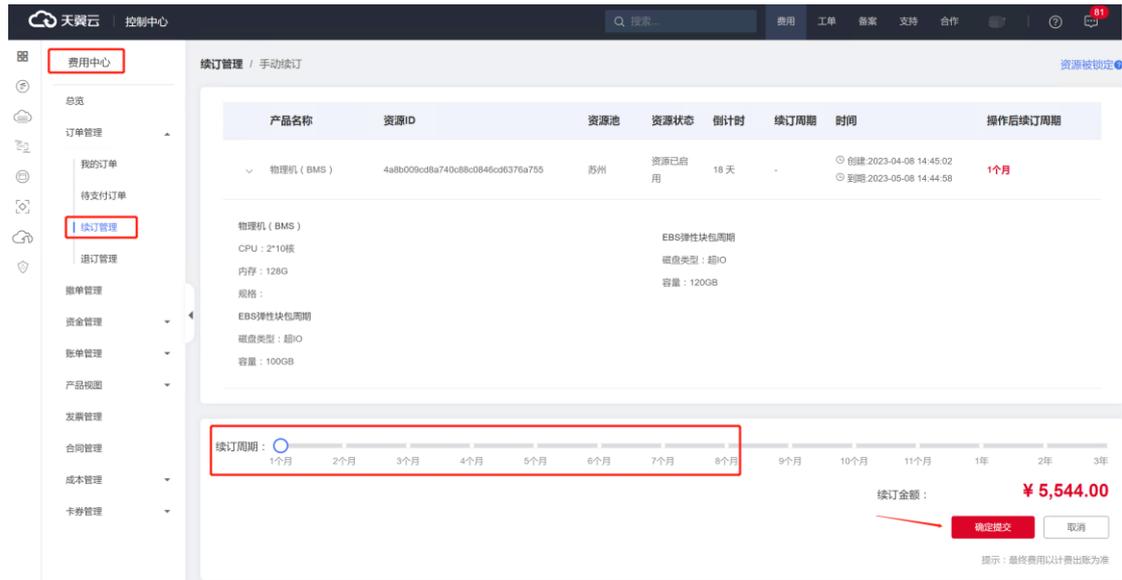
通过业务受理开通的用户可通过业务受理完成续订。



自动续订：用户购买物理机后，在【费用中心】→【续订管理】，选中产品可开通自动续订。



设置续费周期，确认提交即可。



2.5 物理机退订

本章节主要介绍天翼云物理机的退订说明。

退订

- 1、登录天翼云控制中心；
- 2、在左侧导航栏，单击【计算 > 物理机服务】；

3、购买成功后的物理机，在【物理机列表】选择需要退订的物理机，单击【更多】选择退订，并单击【退订】跳转到【管理中心】-【产品中心】退订界面。



4、核实退订的物理机信息，点击退订即可。



2.6 物理机到期处理

本章节主要介绍天翼云物理机到期处理说明。

在物理机资源将要到期时，天翼云会以邮件的方式通知客户，如用户需继续使用，则可联系客户经理执行续费操作或自行在控制台进行续订。如果用户未执行续费操作，天翼云将发送资源超期提醒邮件，并在之后释放物理机资源。

3 快速入门

3.1 入门指引

本章节主要介绍天翼云物理机的入门指引。



3.2 准备工作

本章节主要介绍开通天翼云物理机前的准备工作。

注册天翼云并实名认证

如果您已有一个天翼云账号，请跳到下一个任务。如果您还没有天翼云账号，请参考以下步骤创建。

1. 打开[天翼云官网](#)，单击“注册”。
2. 根据提示信息完成天翼云账号注册，详细操作请参见[如何注册天翼云账号](#)。
3. 参考[实名认证](#)完成个人或企业账号实名认证。

为账户充值

您需要确保账户有足够金额。

- 关于物理机服务器价格，请参见[价格说明](#)。
- 关于充值，请参见[如何给天翼云账户充值](#)。

3.3 开通物理机服务

3.3.1 自助开通

本章节主要介绍如何自助开通天翼云物理机。

1. 登录天翼云控制中心。
2. 在服务列表页面，单击“计算 > 物理机服务”。
3. 在“物理机”界面，单击“申请物理机”。
4. 选择“区域”。

不同区域的云服务产品之间内网互不相通。建议您选择最靠近您业务的区域，这样可以减少网络时延、提高访问速度。

5. 选择“可用区”。

可用区指在同一区域下，电力、网络隔离的物理区域，可用区之间内网互通，不同可用区之间物理隔离。

- 如果您需要提高应用的高可用性，建议您将物理机创建在不同的可用区。
- 如果您需要较低的网络时延，建议您将物理机创建在相同的可用区。

6. 选择“规格”。

包括规格名称、CPU、内存、本地磁盘和扩展配置。当您选择规格列表中的一个规格后，列表下方会展示该规格的名称、组网、用途等信息，以便您根据业务场景进行选择。

其中，扩展配置描述了所选机型的网卡信息。例如：2 x 2*10GE 表示 1 块双网口的 10GE 网卡接入 VPC 网络，1 块双网口的 10GE 扩展网卡支持物理机间的高速互联。

规格中的 CPU、内存、本地磁盘等配置为固定值，不可更改。

不同规格的物理机带宽能力不同，请您根据实际业务慎重选择。

某些规格的物理机支持快速发放能力，选择后，界面会提供“系统盘”参数（在“磁盘”配置项中展示），物理机操作系统直接安装在云硬盘，实现快速发放。

7. 选择“镜像”。

- 公共镜像

常见的标准操作系统镜像，所有用户可见，包括操作系统以及预装的公共应用（SDI 卡驱动、bms-network-config 网络配置程序、cloud-init 初始化工具等）。请根据您的实际需要自助配置应用环境或相关软件。

- 私有镜像

用户基于外部镜像文件或物理机创建的个人镜像，仅用户自己可见。包含操作系统、预装的公共应用以及用户的私有应用。选择私有镜像创建物理机，可以节省您重复配置物理机的时间。

- 共享镜像

您将接受公有云其他用户共享的私有镜像，作为自己的镜像进行使用。

8. 选择“许可类型”。

在云平台上使用操作系统或软件的许可证类型。如果您选择的镜像为免费的，则系统不会展示该参数。如果您选择的镜像为计费镜像，此时系统会展示该参数。

- 使用平台许可证

使用云平台提供的许可证，申请许可证需要支付一定的费用。

- 使用自带许可证（BYOL）

使用用户已有操作系统的许可证，无需重新申请。

9. 设置“磁盘”。

有 SDI 卡的物理机可以使用云硬盘，

磁盘包括系统盘和数据盘。您可以为物理机添加多块数据盘，系统盘大小可以根据需要自定义。

- 系统盘

如果选择支持快速发放的规格，界面会提供系统盘配置项，可以根据需要设置磁盘类型和大小。

- 数据盘

您可以为物理机添加多块数据盘，并设置数据盘的共享功能。

注意：

物理机仅支持 SCSI 磁盘模式。

共享盘：勾选后，数据盘为共享云硬盘。该共享盘可以同时挂载给多台物理机使用。

10. 设置“网络”，包括“虚拟私有云”、“安全组”、“网卡”等信息。

第一次使用云服务时，系统将自动为您创建一个默认的虚拟私有云，包括安全组、网卡。其中，默认虚拟私有云支持的地址范围为 192.168.1.0/24，子网网关为 192.168.1.1，并为子网开启 DHCP 功能。

参数	解释
虚拟私有云	您可以选择使用已有的虚拟私有云，或者单击“查看虚拟私有云”创建新的虚拟私有云。

参数	解释
安全组	<p>安全组用来实现安全组内和安全组间物理机的访问控制，加强物理机的安全保护。用户可以在安全组中定义各种访问规则，当物理机加入该安全组后，即受到这些访问规则的保护。</p> <p>创建物理机时，仅支持选择一个安全组。但是物理机创建成功后，可以为物理机关联多个安全组，配置方法请参见 5.2 配置安全组。</p> <p>说明</p> <p>物理机初始化需要确保安全组出方向规则至少满足如下要求：</p> <ul style="list-style-type: none"> 1 协议：TCP 1 端口范围：80 1 远端地址：169.254.0.0/16 <p>如果您使用的是默认安全组出方向规则，则已经包括了如上要求，可以正常初始化。默认安全组出方向规则为：</p> <ul style="list-style-type: none"> 1 协议：Any 1 端口范围：Any 1 远端地址：0.0.0.0/16
网卡	<p>包括主网卡和扩展网卡。您可以添加多张扩展网卡，并指定网卡（包括主网卡）的 IP 地址。</p> <p>注意</p> <ul style="list-style-type: none"> 1 主网卡用于系统的默认路由，不允许删除。 1 如果您选择自动分配 IP 地址，请不要在物理机发放完成后修改私有 IP 地址，避免和其他物理机 IP 冲突。 1 为网卡分配固定 IP 地址后，不能批量创建物理机。

参数	解释
高速网卡	<p>物理机之间的高速互连网络，为物理机提供更高的带宽。</p> <p>一台物理机最多有两块高速网卡，并且依赖于扩展网卡总带宽（可以在规格的“扩展配置”列查看）。例如，扩展网卡总带宽为 2 * 10GE，如果第一块高速网卡的带宽为 2 * 10GE，那么您不能再添加第二块高速网卡。</p> <p>说明</p> <p>同一台物理机的多张高速网卡不能选择同一个高速网络。</p>
弹性公网 IP	<p>弹性公网 IP 是指将公网 IP 地址和路由网络中关联的物理机绑定，以实现虚拟私有云内的物理机通过固定的公网 IP 地址对外提供访问服务。</p> <p>您可以根据实际情况选择以下三种方式：</p> <p>不使用：物理机不能与外部网络互通，仅可以在私有网络中部署业务或构建集群。</p> <p>自动分配：自动为物理机分配独享带宽的弹性公网 IP，带宽值由您设定。</p> <p>使用已有：为物理机分配已有弹性公网 IP。</p> <p>说明</p> <p>选择已有弹性公网 IP 后，不能批量创建物理机。</p>
规格	电信
计费方式	<p>弹性公网 IP 选择“自动分配”时，需配置该参数。</p> <p>按带宽计费：指定带宽上限，按使用时间计费，与使用的流量无关。</p> <p>按流量计费：按照实际使用的流量来计费。</p>
带宽	<p>弹性公网 IP 选择“自动分配”时，需配置该参数。</p> <p>带宽大小，单位 Mbit/s。</p>

11. 设置物理机登录方式。

“密钥对”方式创建的物理机安全性更高，建议选择“密钥对”方式。如果您习惯使用“密码”方式，请增强密码的复杂度，如下表所示，保证密码符合要求，防止被恶意攻击。

- 密钥对

指使用密钥对作为登录物理机的鉴权方式。您可以选择使用已有的密钥，或者单击“查看密钥对”创建新的密钥。

如果选择使用已有的密钥，请确保您已在本地获取该文件，否则，将影响您正常登录物理机。

- 密码

指使用设置初始密码方式作为物理机的鉴权方式，此时，您可以通过用户名密码方式登录物理机。Linux 操作系统时为 root 用户的初始密码，Windows 操作系统时为 Administrator 用户的初始密码。密码复杂度需满足要求。

注意：

Windows 操作系统不支持选择密码登录方式。

参数	规则	样例
密码	密码长度范围为 8 到 26 位。 密码至少包含以下 4 种字符中的 3 种： 大写字母 小写字母 数字 特殊字符，包括!@\$%^_+=[]{}:;./? 密码不能包含用户名或用户名的逆序。 Windows 系统的物理机，不能包含用户名中超过两个连续字符的部分。	Test12\$@

12. 设置“物理机名称”。

名称可自定义，但需符合命名规则：只能由中文字符、英文字母、数字及“_”、“-”、“.”组成。

一次创建多台物理机，系统会自动按序增加后缀。例如：输入 bms，服务器名称为 bms-0001，bms-0002，……。再次购买多台服务器时，命名从上次最大值连续增加，例如：输入 bms，已有服务器 bms-0010，新创服务器名称为 bms-0011、bms-0012、……，命名达到 9999 时，从 0001 开始。

设置您创建的物理机数量，最多为 24 台。

设置完成后，您可以通过单击“价格计算器”，查询当前配置的费用。

说明：

- 当配额充足时，一次最多可以创建 24 台物理机；若配额不足 24 台，一次最多可创建数量为配额余量。
- 在设置网络信息时，若您选择自定义网卡或高速网卡的 IP 地址，或者选择已有弹性公网 IP，则一次只能创建一台物理机。

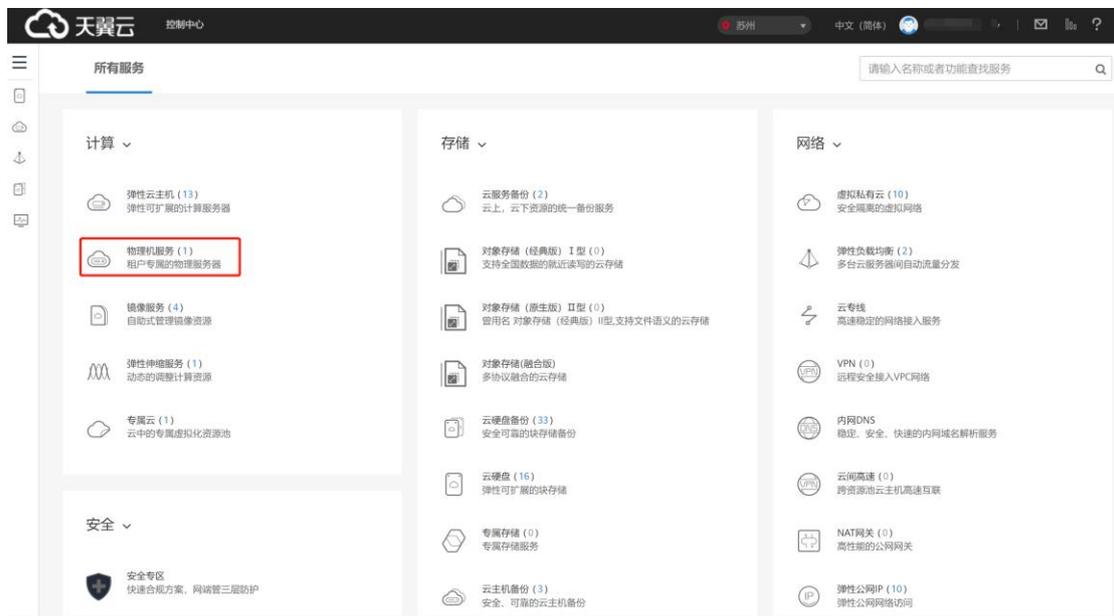
13. 单击“立即申请”。

14. 在确认规格页面，单击“提交”。

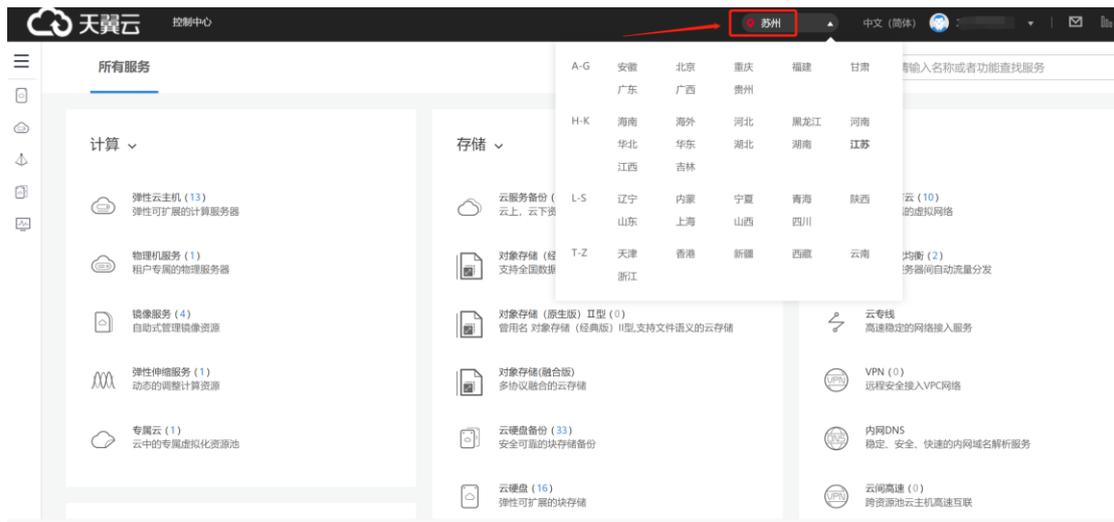
3.3.2 通过客户经理开通

用户开通项目制物理机请通过客户经理联系业务受理开通。

1. 用户登陆控制中心后，单击“计算 > 物理机服务”；



2. 单击管理控制中心的区域按钮，选择地域和项目。



3. 选择“计算 > 物理机服务”，进入物理机页面。如下图所示，页面提示通过拨打 400 电话联系客服开通物理机服务，联系客服，或可直接联系客户所对应的客户经理开通物理机服务。



4. 客户将具体的物理机需求（如规格、数量、网络配置信息等）告知客户经理，由客户经理完成后续的开通流程。

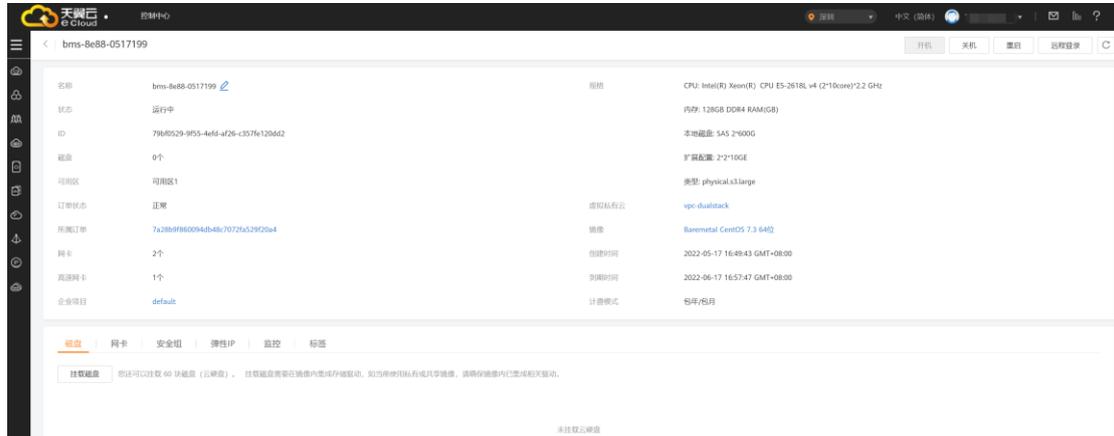
5. 物理机创建需 1-2 个工作日，创建完成后，客户将收到客户经理的通知，并可登录平台进行使用。

3.4 查看物理机详情

本章节主要介绍如何查看天翼云物理机详情。

在您申请了物理机后，可以通过管理控制台查看和管理您的物理机。在物理机列表页面点击

物理机名称，即可跳转详情页面查看物理机详细信息，如：名称、状态、规格、VPC 等信息。还可以在此页面执行挂载、卸载云硬盘、绑定、解绑定弹性 IP 等。如下图所示。



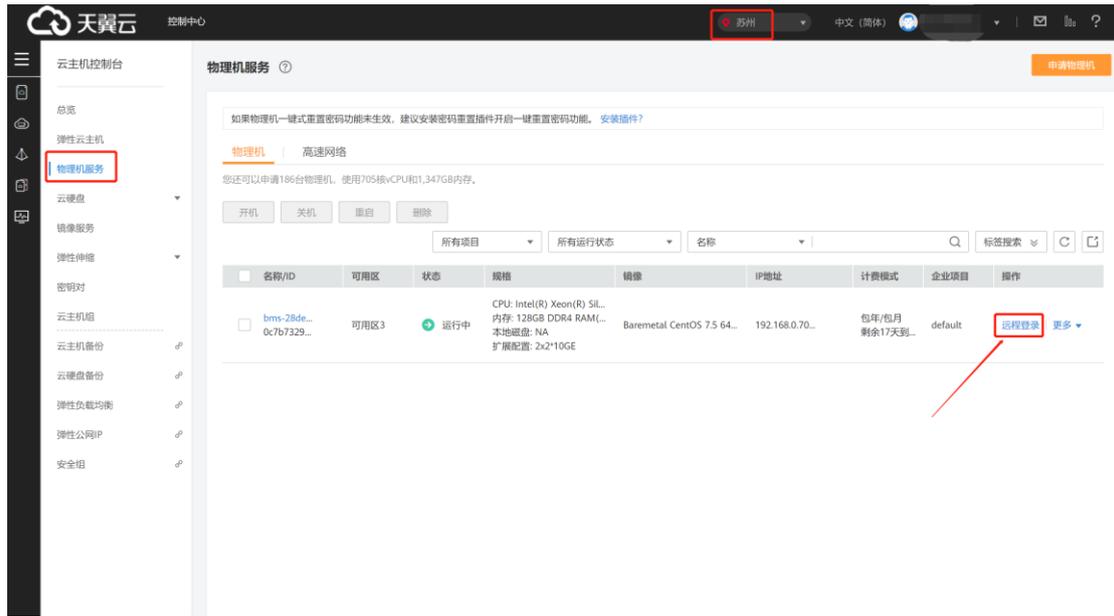
3.5 登录物理机服务

3.5.1 登录 linux 物理机

3.5.1.1 VNC 登录物理机：

本章节主要介绍了 VNC 登录天翼云物理机。

1. 登录天翼云控制中心，选择区域和项目。
2. 单击【服务列表】，选择【物理机服务】进入物理机列表页。
3. 选择要登录的物理机所在行的【远程登录】；



4. 在弹出的页面中，输入创建物理机时设置的密码。

5. 还可点击按钮执行相应操作，如 Send CtrlAltDel。

3.5.1.2 SSH 密码方式登录

本章节主要介绍天翼云物理机 SSH 密码登录方式。

SSH 密码方式登录

只有“运行中”的物理机才允许用户登录。

登录 Linux 物理机的方式有：

SSH 方式登录

仅适用于 Linux 物理机，您可以使用远程登录工具（例如 PuTTY），登录物理机。此时，需要该物理机绑定弹性 IP。

远程登录方式

未绑定弹性 IP 的物理机可通过管理控制台提供的远程登录方式直接登录，仅适用于 Linux 物理机。

前提条件：

物理机已经绑定弹性 IP 地址。

- 本地使用 Windows 操作系统

如果您用于登录物理机的系统是 Windows，以 PuTTY 为例，可以按照下面方式登录物理机。

1. 运行 PuTTY。
2. 单击“Session”，在“Host Name (or IP address)”下的输入框中输入物理机的弹性 IP 地址，连接方式选择 SSH。
3. 单击“Open”。
4. 输入用户名“root”和密码，登录物理机。

- 本地使用 Linux 操作系统

如果您登录物理机的系统是 Linux，您可以在服务器的命令行中运行如下命令登录物理机。

```
ssh "{物理机的弹性 IP 地址}"
```

3.5.1.3 SSH 密钥方式登录

本章节主要介绍天翼云物理机 SSH 密钥方式登录

前置条件:

物理机已经绑定弹性 IP 地址。

已在本地获取登录物理机的私钥文件。

本地使用 Windows 操作系统

如果您本地使用 Windows 操作系统登录 Linux 物理机，可以按照下面方式登录。

我们以 PuTTY 工具为例介绍如何登录 Linux 物理机。使用 PuTTY 登录物理机前，需要先将私钥文件转化为 .ppk 格式。

1. 判断私钥文件是否为 .ppk 格式。

是，执行步骤 7。否，执行步骤 2。

- 在以下路径中下载 PuTTY 和 PuTTYgen。

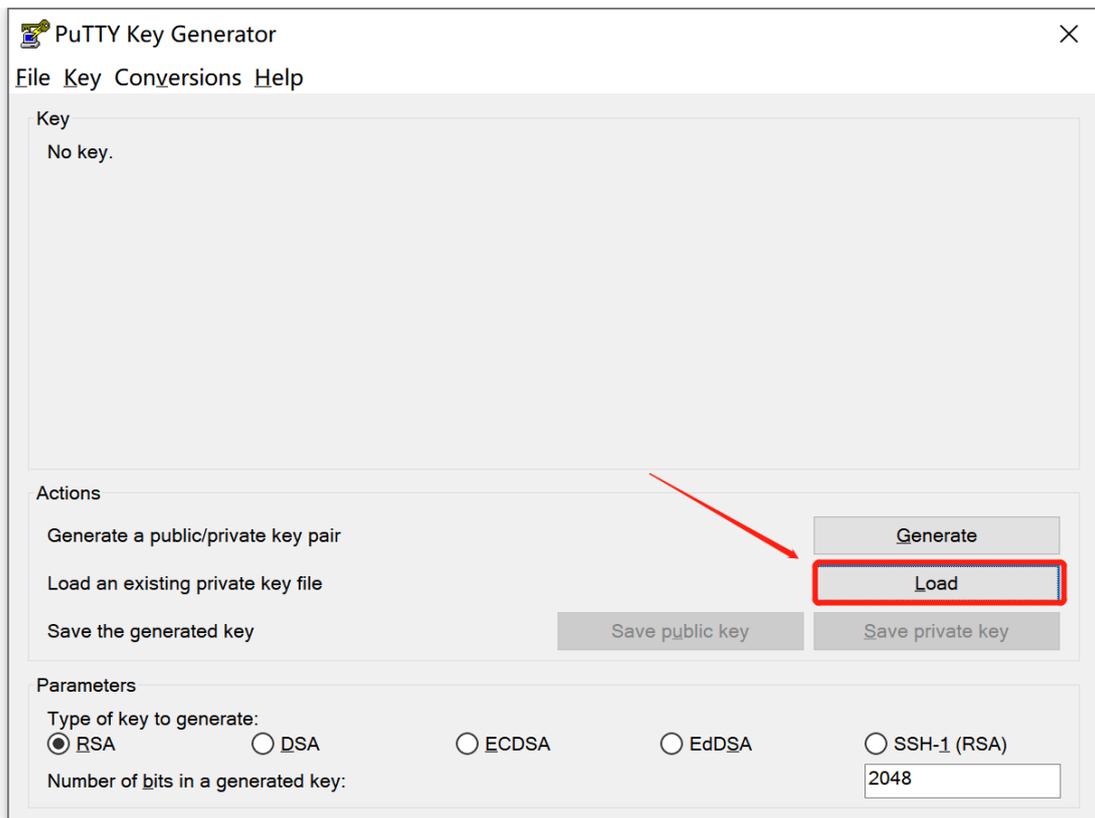
<http://www.chiark.greenend.org.uk/~sgtatham/putty/latest.html>

说明：

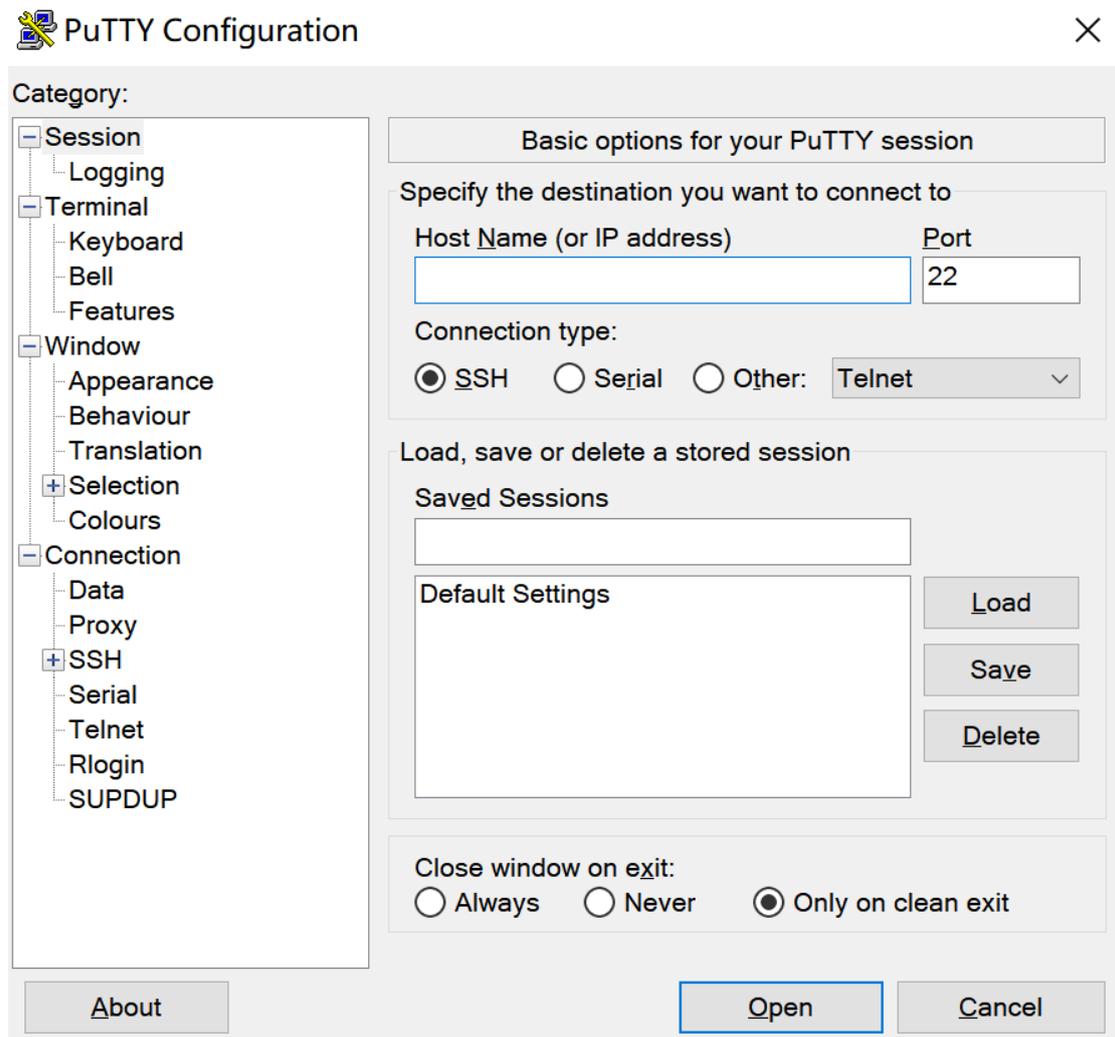
PuTTYgen 是密钥生成器，用于创建密钥对，生成一对公钥和私钥供 PuTTY 使用。

- 运行 PuTTYgen。
- 在“Actions”区域，单击“Load”，并导入创建物理机时保存的私钥文件。

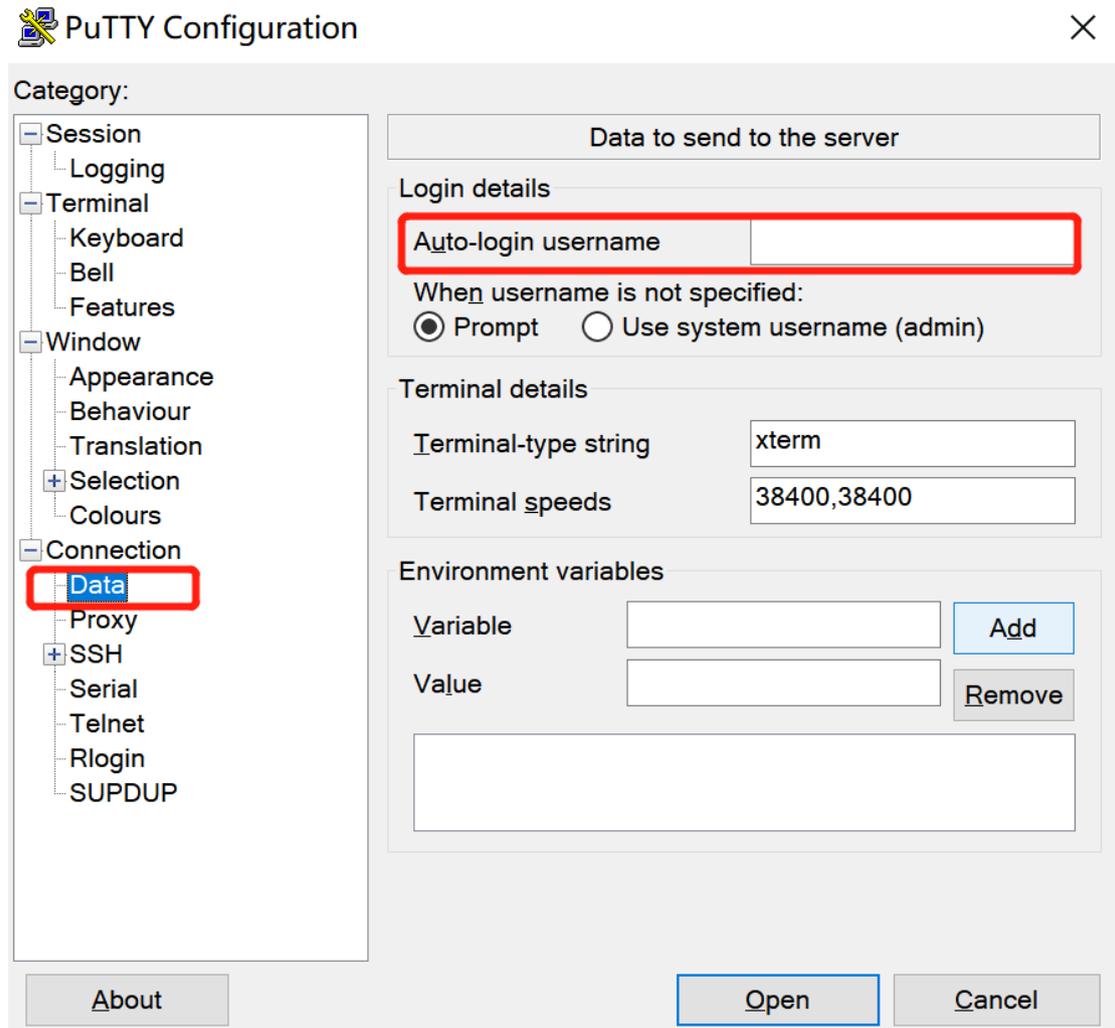
导入时注意确保导入的格式要求为“All files (*.*)”。



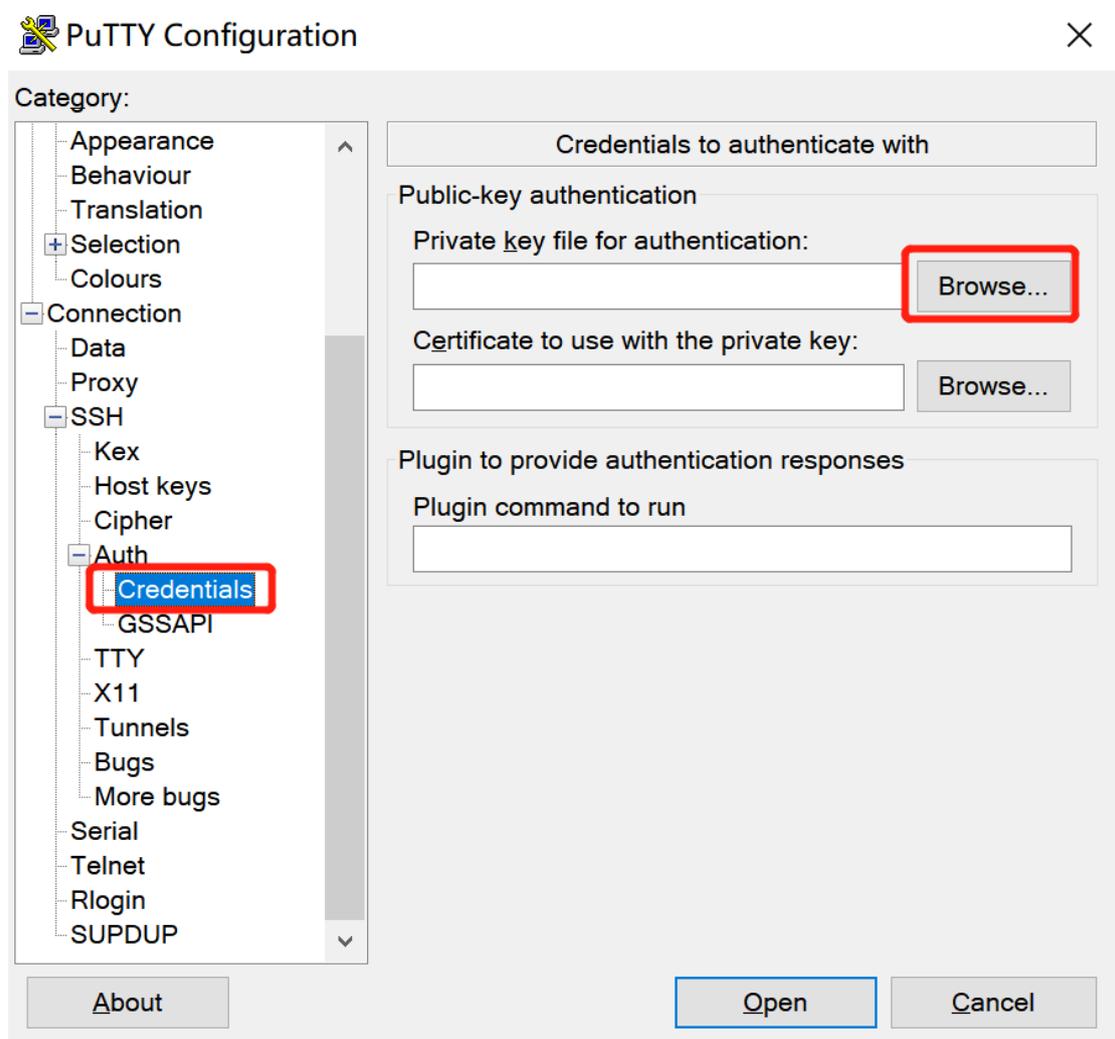
- 单击“Save private key”。
- 保存转化后的私钥到本地。例如：kp-123.ppk
- 双击“PUTTY.EXE”，打开“PuTTY Configuration”。如下图，



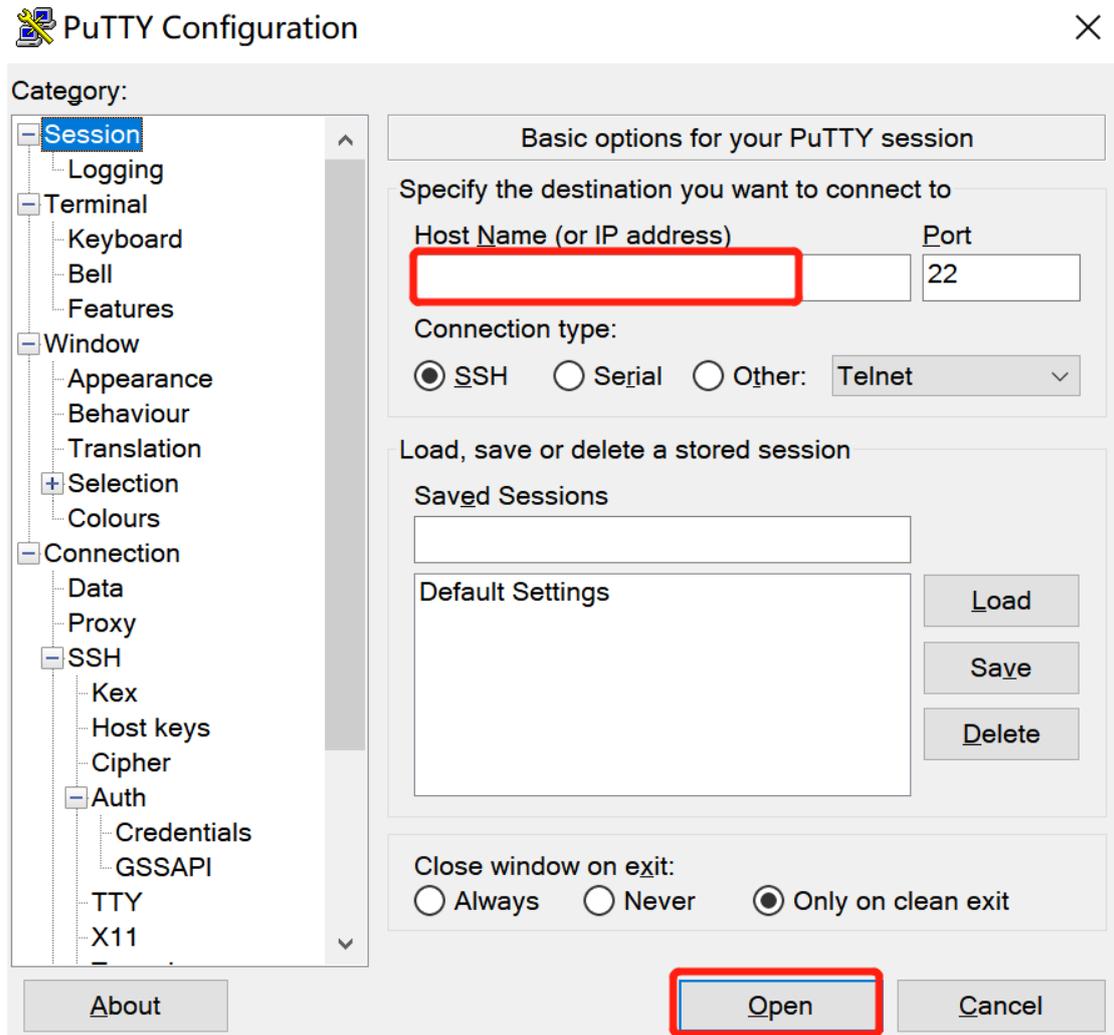
8. 选择“Connection > Data”，在 Auto-login username 处输入镜像的用户名。



9. 选择“Connection > SSH > Auth”，在最下面一个配置项“Private key file for authentication”中，单击“Browse”，选择步骤6转化的密钥。



10. 单击“Session”，在“Host Name (or IP address)”下的输入框中输入物理机的弹性 IP 地址。



11. 单击“Open”，登录物理机。

本地使用 Linux 操作系统

如果您本地使用 Linux 操作系统登录 Linux 物理机，可以按照下面方式登录。下面步骤以私钥文件是“KeyPair-ee55”为例进行介绍。

1. 在您的 linux 计算机的命令行中执行如下命令，变更权限。

```
chmod 600 /path/KeyPair-ee55
```

说明：

- 上述命令的 path 为密钥文件的存放路径。

2. 执行如下命令登录物理机。

```
ssh -i /path/KeyPair-ee55 xxx@ 物理机弹性 IP 地址
```

说明：

- path 为密钥文件的存放路径。
- xxx 为物理机下发过程中默认创建的，如果创建物理机时使用 Red Hat 操作系统，则用户名为 rhel。

3.5.1.4 远程登录方式

约束与限制：

- 仅 Linux 物理机支持远程登录。
- 只有该物理机的创建者，或者拥有 Tenant Administrator 或 Server Administrator 角色的用户才能远程登录该物理机。
- 物理机远程登录的界面不支持中文输入法，不支持桌面图形化操作。
- 物理机远程登录，对 Ctrl、Alt 等快捷键支持不够友好，比如“Alt + ASCII 码”表示的特殊字符会输出多个特殊字符。

密钥对（KeyPair）登录方式创建的 Linux 物理机，需要先使用密钥登录物理机（参考 [SSH 密钥方式登录](#)）设置密码才可以进行远程登录。具体操作如下：

使用密钥登录物理机后切换到 root 帐户，执行命令 passwd 即可设置 root 密码。

```
[root@serverc28ef36e-08ef-4d94-8921-155fa4d4332b ~]# passwd
Changing password for user root.
New password:
Retype new password:
passwd: all authentication tokens updated successfully.
[root@serverc28ef36e-08ef-4d94-8921-155fa4d4332b ~]#
```

1. 登录管理控制台。
2. 单击管理控制台左上角的  ，选择地域和项目。
3. 选择“计算 > 物理机”，进入物理机页面。

4. 选择一台物理机，单击“远程登录”。开始建立连接，大约 1 分钟后进入登录界面，按“Enter”后输入用户名和密码即可登录。

说明：

- 获取远程登录链接后，如果 10 分钟内未登录，则该链接失效，需要重新获取。
- 远程登录的界面如果 10 分钟未操作，则该页面失效，需要重新获取链接并登录。
- 如果在登录界面按“Enter”后没有响应，可能因为该物理机所使用的镜像未配置远程登录功能，请参考[远程登录物理机时界面操作无响应，如何解决？](#)解决。
- 远程登录物理机后，如果控制台显示异常（比如错行、乱码），请参考[远程登录物理机后控制台显示异常，如何解决？](#)解决。

3.5.2 登录 windows 物理机

本章节主要介绍如何登录 windows 天翼云物理机。

登录方式说明

只有“运行中”的物理机才允许用户登录。

登录 Windows 物理机的方式为：MSTSC 密码方式，仅适用于 Windows 物理机，您可以通过在本机运行 MSTSC 方式登录物理机。此时，需要该物理机绑定弹性公网 IP。

MSTSC 密码方式登录

我们以 mstsc.exe 远程登录工具为例介绍如何登录物理机。

1. 在 Windows 物理机下发并配置完成之后，在管理控制台页面对应的 Windows 物理机所在行，单击“更多 > 获取密码”，选择本地下发物理机的私钥文件。然后单击“获取密码”，即可获取 Windows 物理机的用户名和密码。
2. 在本地 Windows 操作系统“开始”菜单中输入“mstsc.exe”，在远程桌面连接的界面输入 Windows 物理机的弹性 IP 以及用户名，并按照提示输入步骤 1 中获取的密码即可完成登录。

3.5.3 初始化数据盘

本章节主要介绍天翼云物理机如何初始化数据盘。

以下场景，需要登录物理机初始化磁盘后，磁盘才能正常使用。

通过物理机管理控制台创建物理机时，直接创建了数据盘，此时需要初始化该数据盘。

通过云硬盘管理控制台创建了数据盘，然后将该数据盘挂载至物理机，此时需要初始化该数据盘。

本文以“CentOS 7.0 64bit”为例进行介绍。不同操作系统的物理机格式化操作可能不同，本文仅供参考，具体步骤和差异请参考对应的物理机操作系统的产品文档。

注意

首次使用云硬盘时，如果您未参考本章节对云硬盘执行初始化操作，主要包括创建分区和文件系统等操作，那么当后续扩容云硬盘时，新增容量部分的磁盘可能无法正常使用。

前提条件

已以“root”帐户登录物理机。

已挂载数据盘至物理机，且该数据盘未初始化。

3.5.3.1 划分分区并挂载磁盘

本操作以该场景为例，当物理机挂载了一块新的数据盘时，需要将该数据盘设为主分区，文件系统设为 ext4 格式，挂载在“/mnt/sdc”目录下，并设置开机启动自动挂载。

1. 执行以下命令，查看新增数据盘。

```
fdisk -l
```

回显类似如下信息：

```
[root@ecs-b656 test]# fdisk -l

Disk /dev/sda: : 42.9 GB, 42949672960 bytes, 83886080 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk label type: dos
Disk identifier: 0x000cc4ad

Device Boot          Start      End          Blocks      Id  System
/dev/sda1    *          2048     2050047     1024000     83  Linux
/dev/sda2                2050048     22530047     10240000     83  Linux
/dev/sda3                22530048     24578047     1024000     83  Linux
/dev/sda4                24578048     83886079     29654016     5   Extended
/dev/sda5                24580096     26628095     1024000     82  Linux swap /
Solaris

Disk /dev/sdb: : 10.7 GB, 10737418240 bytes, 20971520 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
```

表示当前的物理机有两块磁盘，“/dev/sda”是系统盘，“/dev/sdb”是新增数据盘。

2. 执行以下命令，进入 fdisk 分区工具，开始对新增数据盘执行分区操作。

```
fdisk 新增数据盘
```

以新挂载的数据盘“/dev/sdb”为例：

```
fdisk /dev/sdb
```

回显类似如下信息：

```
[root@ecs-b656 test]# fdisk /dev/sdb
Welcome to fdisk (util-linux 2.23.2).
Changes will remain in memory only, until you decide to write them.
Be careful before using the write command.
Device does not contain a recognized partition table
Building a new DOS disklabel with disk identifier 0xb00005bd.
Command (m for help):
```

3. 输入“n”，按“Enter”，开始新建分区。

回显类似如下信息：

```
Command (m for help): n
Partition type:
   p   primary (0 primary, 0 extended, 4 free)
   e   extended
```

表示磁盘有两种分区类型：

“p”表示主要分区。

“e”表示延伸分区。

4. 以创建一个主要分区为例，输入“p”，按“Enter”，开始创建一个主分区。

回显类似如下信息：

```
Select (default p): p
Partition number (1-4, default 1):
```

“Partition number”表示主分区编号，可以选择1-4。

5. 以分区编号选择“1”为例，输入主分区编号“1”，按“Enter”。

回显类似如下信息：

```
Partition number (1-4, default 1): 1
First sector (2048-20971519, default 2048):
```

“First sector”表示初始磁柱区域，可以选择 2048–20971519，默认为 2048。

6. 以选择默认初始磁柱编号 2048 为例，按 “Enter”。

回显类似如下信息：

```
First sector (2048-20971519, default 2048):  
Using default value 2048  
Last sector, +sectors or +size {K,M,G} (2048-20971519, default  
20971519):
```

“Last sector”表示截止磁柱区域，可以选择 2048–20971519，默认为 20971519。

7. 以选择默认截止磁柱编号 20971519 为例，按 “Enter”。

回显类似如下信息：

```
Last sector, +sectors or +size {K,M,G} (2048-20971519, default  
20971519):  
Using default value 20971519  
Partition 1 of type Linux and of size 10 GiB is set  
Command (m for help):
```

表示分区完成，即为 10GB 的数据盘新建了 1 个分区。

8. 输入 “p”，按 “Enter”，查看新建分区的详细信息。

回显类似如下信息：

```
Command (m for help): p  
Disk /dev/sdb: 10.7 GB, 10737418240 bytes, 20971520 sectors  
Units = sectors of 1 * 512 = 512 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk label type: dos  
Disk identifier: 0xb00005bd
```

```
Device
Boot          Start                End                Blocks          Id  Syst
em
/dev/sdb1          2048                20971519          10484736
83  Linux

Command (m for help):
```

表示新建分区“/dev/sdb1”的详细信息。

9. 输入“w”，按“Enter”，将分区结果写入分区表中。

回显类似如下信息：。

```
Command (m for help): w
The partition table has been altered!

Calling ioctl() to re-read partition table.
Syncing disks.
```

表示分区创建完成。

10. 执行以下命令，将新的分区表变更同步至操作系统。

```
partprobe
```

11. 执行以下命令，将新建分区文件系统设为系统所需格式。

```
mkfs -t 文件系统格式 /dev/sdb1
```

以设置文件系统为“ext4”为例：

```
mkfs -t ext4 /dev/sdb1
```

回显类似如下信息：

```
[root@ecs-b656 test]# mkfs -t ext4 /dev/sdb1
mke2fs 1.42.9 (28-Dec-2013)
Filesystem label=
OS type: Linux
Block size=4096 (log=2)
Fragment size=4096 (log=2)
Stride=0 blocks, Stripe width=0 blocks
655360 inodes, 2621184 blocks
131059 blocks (5.00%) reserved for the super user
First data block=0
Maximum filesystem blocks=2151677952
80 block groups
32768 blocks per group, 32768 fragments per group
8192 inodes per group
Superblock backups stored on blocks:
    32768, 98304, 163840, 229376, 294912, 819200, 884736,
1605632

Allocating group tables: done
Writing inode tables: done
Creating journal (32768 blocks): done
Writing superblocks and filesystem accounting information: done
格式化需要等待一段时间，请观察系统运行状态，不要退出。
```

注意

不同文件系统支持的分区个数和分区大小不同，请根据您的业务需求选择合适的文件系统。

12. 执行以下命令，新建挂载点。

mkdir 挂载点

以新建挂载点“/mnt/sdc”为例：

```
mkdir /mnt/sdc
```

13. 执行以下命令，将新建分区挂载到 步骤 12 中新建的挂载点下。

```
mount /dev/sdb1 挂载点
```

以挂载新建分区至 “/mnt/sdc” 为例：

```
mount /dev/sdb1 /mnt/sdc
```

14. 执行以下命令，查看挂载结果。

```
df -TH
```

回显类似如下信息：

```
[root@ecs-b656 test]# df -TH
Filesystem      Type          Size  Used Avail Use% Mounted on
/dev/sda2       xfs           11G   7.4G  3.2G   71% /
devtmpfs        devtmpfs     4.1G     0   4.1G    0% /dev
tmpfs           tmpfs        4.1G   82k   4.1G    1%
/dev/shm
tmpfs           tmpfs        4.1G   9.2M   4.1G    1% /run
tmpfs           tmpfs        4.1G     0   4.1G    0%
/sys/fs/cgroup
/dev/sda3       xfs           1.1G   39M   1.1G    4% /home
/dev/sda1       xfs           1.1G  131M   915M   13% /boot
/dev/sdb1       ext4          11G   38M   9.9G    1% /mnt/sdc
```

表示新建分区 “/dev/sdb1” 已挂载至 “/mnt/sdc” 。

3.5.3.2 设置开机自动挂载磁盘

如果您需要在物理机系统启动时自动挂载磁盘，不能采用在/etc/fstab 直接指定 /dev/sdb1 的方法，因为云中设备的顺序编码在关闭或者开启物理机过程中可能发生改变，例如/dev/sdb1 可能会变成/dev/sdb2。推荐使用 UUID 来配置自动挂载数据盘。

1. 执行如下命令，查询磁盘分区的 UUID。

`blkid` 磁盘分区

以查询磁盘分区“/dev/sdb1”的 UUID 为例：

```
blkid /dev/sdb1
```

回显类似如下信息：

```
[root@ecs-b656 test]# blkid /dev/sdb1
/dev/sdb1: UUID="1851e23f-1c57-40ab-86bb-5fc5fc606ffa" TYPE="ext4"
```

表示“/dev/sdb1”的 UUID。

2. 执行以下命令，使用 vi 编辑器打开“fstab”文件。

```
vi /etc/fstab
```

3. 按“i”，进入编辑模式。
4. 将光标移至文件末尾，按“Enter”，添加如下内容。

```
UUID=1851e23f-1c57-40ab-86bb-5fc5fc606ffa /mnt/sdc          ext4 defaults          0          2
```

5. 按“ESC”后，输入:wq，按“Enter”。

保存设置并退出编辑器。

3.5.4 安装一键式重置密码插件

公有云平台提供了一键式重置密码功能。物理机的密码丢失或过期时，如果您的物理机提前安装了一键式重置密码插件“Cloud-reset-pwd Agent”，则可以应用一键式重置密码功能，为物理机设置新密码。

该方法方便、有效，建议您成功创建、登录物理机后，安装密码重置插件，具体操作请参见本节内容。

说明：

- 如果您使用的公共镜像已安装重置密码插件，或者您在制作私有镜像时已安装该插件，则无需重复安装。如何检查物理机是否已安装该插件，请参见“Linux 物理机安装方法”中的步骤 1 或“Windows 物理机安装方法”中的步骤 1。
- 如果您需要卸载该插件，可参考 Linux 物理机卸载方法或 Windows 物理机卸载方法进行卸载。

安装须知：

1. 该插件为物理机提供的密码重置插件，仅适用于物理机，不能用于弹性云主机。
另外请不要使用除物理机提供的插件以外的其他密码重置插件。
2. 用户自行决定是否安装“Cloud-reset-pwd Agent”插件，使物理机具备一键式重置密码功能。
3. 安装完成后，禁止自行卸载插件，因为可能导致管理控制台判断失误，从而无法完成密码重置。
4. 重装物理机操作系统后，一键式重置密码功能失效。如需继续使用该功能，请重新安装“Cloud-reset-pwd Agent”插件。
5. 物理机密码过期或丢失后，再安装“Cloud-reset-pwd Agent”插件，一键重置密码功能不生效。
6. 物理机需绑定弹性 IP，才能自动更新 CloudResetPwdAgent，或者需要您手动下载升级包，完成安装或升级。
7. 一键重置密码插件“CloudResetPwdAgent”已按照 GNU General Public License v2.0 协议开源至Github 平台，开放、透明、安全，请您放心使用。

前提条件：

- 物理机的状态为“运行中”。
- 对于 Windows 物理机，需保证 C 盘可写入，且剩余空间大于 600MB。对于 Linux 物理机，需保证根目录可写入，且剩余空间大于600MB。
- 物理机使用的VPC网络DHCP不能禁用。
- 物理机网络正常通行。
- 物理机安全组出方向规则满足如下要求：
 - 协议：TCP
 - 端口范围：80
 - 远端地址：169.254.0.0/16

如果您使用的是默认安全组出方向规则，则已经包括了如上要求，可以正常初始化。默认安全组出方向规则为：

- 协议：ANY
- 端口范围：ANY
- 远端地址：0.0.0.0/16

3.5.4.1 Linux 物理机安装方法

本章节主要介绍如何安装天翼云 linux 物理机重置密码插件。

步骤 1 检查物理机服务是否已安装密码重置插件 CloudResetPwdAgent 和 CloudResetPwdUpdateAgent。

1.1. 登录物理机服务。

1.2. 检查是否已安装 CloudrResetPwdAgent。检查方法如下：

- ①. 确认物理机服务的根目录下，存在 CloudrResetPwdAgent 目录。
- ②. 执行以下命令，确认 CloudResetPwdAgent 的状态不是“unrecognized service”。

```
service cloudResetPwdAgent status
```

检查结果同时满足以上两个要求，表示物理机服务已安装插件 CloudResetPwdAgent。

- 是，执行步骤 1.3。
- 否，执行步骤 2。

1.3. 检查是否已安装 CloudResetPwdUpdateAgent。检查方法如下：

①. 确认物理机服务的根目录下，存在 CloudResetPwdUpdateAgent 目录。

②. 执行以下命令，确认 CloudResetPwdUpdateAgent 的状态不是 “unrecognized service”。

```
service cloudResetPwdUpdateAgent status
```

检查结果同时满足以上两个要求，表示物理机服务已安装插件 CloudResetPwdUpdateAgent。

- 是，结束。
- 否，执行步骤 2。

步骤 2 下载一键式重置密码插件 CloudResetPwdAgent 和 CloudResetPwdUpdateAgent。

下载并解压软件包 CloudResetPwdAgent.zip。

下载地址：<http://oos-cn.ctyunapi.cn/downfile/2020%20download/CloudResetPwdAgent-Linux.zip>

步骤 3 安装一键式重置密码插件 CloudResetPwdAgent 和 CloudResetPwdUpdateAgent。

3.1 将步骤 2 中解压后的文件放置在物理机服务的同一目录下，否则会安装失败。

说明：安装插件对解压后文件在物理机服务的具体放置目录无特殊要求，只需保证存放在同一目录下即可。

3.2 执行以下命令，进入文件 CloudResetPwdUpdateAgent.Linux。

```
cd CloudResetPwdUpdateAgent.Linux
```

3.3 执行以下命令，添加文件 setup.sh 的运行权限。

```
chmod +x setup.sh
```

3.4 执行以下命令，安装插件。

```
sudo sh setup.sh
```

3.5 执行以下命令，检查密码重置插件是否安装成功。

```
service cloudResetPwdAgent status
```

```
service cloudResetPwdUpdateAgent status
```

如果服务 CloudResetPwdAgent 和 CloudResetPwdUpdateAgent 的状态均不是“unrecognized service”，表示插件安装成功，否则安装失败。

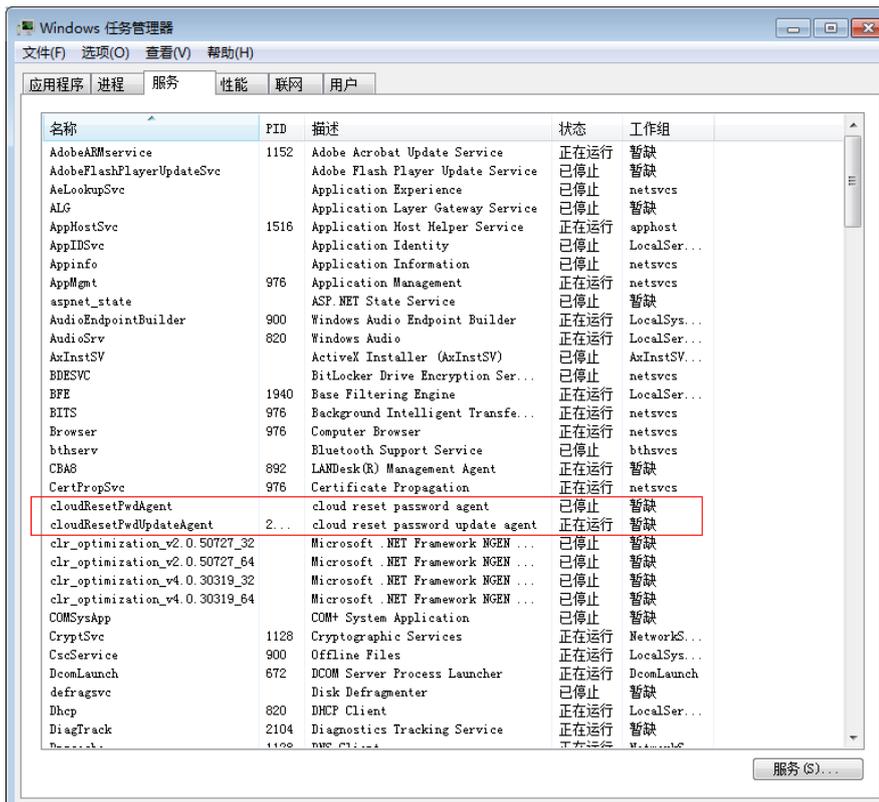
说明：如果密码重置插件安装失败，请检查安装环境是否符合要求，并重试安装操作。

3.5.4.2 Windows 物理机服务安装方法

本章节主要介绍如何安装天翼云 windows 物理机重置密码插件。

步骤 1 检查物理机服务是否已安装密码重置插件 CloudResetPwdAgent 和 CloudResetPwdUpdateAgent。检查方法如下：

查看任务管理器，如果找到 cloudResetPwdAgent 服务和 CloudResetPwdUpdateAgent 服务，如下图所示，表示物理机服务已安装密码重置插件。



是，结束。

否，执行步骤 2。

步骤 2 下载一键式重置密码插件 CloudResetPwdAgent 和 CloudResetPwdUpdateAgent。

下载并解压软件包 CloudResetPwdAgent.zip。

下载地址：[http://oos-](http://oos-cn.ctyunapi.cn/downfile/2020%20download/CloudResetPwdAgent%20-Windows.zip)

[cn.ctyunapi.cn/downfile/2020%20download/CloudResetPwdAgent%20-Windows.zip](http://oos-cn.ctyunapi.cn/downfile/2020%20download/CloudResetPwdAgent%20-Windows.zip)

步骤 3 安装一键式重置密码插件 CloudResetPwdAgent 和 CloudResetPwdUpdateAgent。

3.1 将步骤 2 中解压后的文件放置在物理机服务的同一目录下，否则会安装失败。

说明：安装插件对解压后文件在物理机服务的具体放置目录无特殊要求，只需保证存放在同一目录下即可。

3.2 双击 CloudResetPwdUpdateAgent.Windows 目录下的“setup.bat”文件。

安装密码重置插件。

3.3 查看任务管理器，检查密码重置插件是否安装成功。

如果在任务管理器中查找到了 cloudResetPwdAgent 服务和 cloudResetPwdUpdateAgent 服务，表示安装成功，否则安装失败。

说明：如果密码重置插件安装失败，请检查安装环境是否符合要求，并重试安装操作。

3.5.4.3 Linux 物理机服务卸载方法

本章节主要介绍如何卸载天翼云物理机重置密码插件。

登录物理机服务，依次执行以下命令：

1. `sudo bash /CloudrResetPwdAgent/bin/cloudResetPwdAgent.script remove`
2. `sudo bash /CloudResetPwdUpdateAgent/bin/cloudResetPwdUpdateAgent.script remove`
3. `sudo rm -rf /CloudrResetPwdAgent`
4. `sudo rm -rf /CloudResetPwdUpdateAgent`

3.5.4.4 Windows 物理机服务卸载方法

本章节主要介绍如何卸载天翼云物理机重置密码插件。

1. 登录物理机服务器，进入“C:\CloudResetPwdUpdateAgent\bin”目录。
2. 双击“UninstallApp-NT.bat”。
3. 删除“C:\CloudResetPwdUpdateAgent”。
4. 进入“C:\CloudResetPwdAgent\bin”目录。
5. 双击“UninstallApp-NT.bat”。
6. 删除“C:\CloudResetPwdAgent”

3.6 部署应用服务

本章节主要介绍如何在天翼云物理机部署应用服务。

安装及启动Nginx

1. 输入 `yum install nginx` 命令安装 Nginx，当需要确认时输入“y”确认。

```
Installed:
nginx.x86_64 1:1.12.2-3.el7

Dependency Installed:
dejavu-fonts-common.noarch 0:2.33-6.el7
fontconfig.x86_64 0:2.13.0-4.3.el7
gd.x86_64 0:2.0.35-26.el7
libX11.x86_64 0:1.6.5-2.el7
libXau.x86_64 0:1.0.8-2.1.el7
libjpeg-turbo.x86_64 0:1.2.90-6.el7
libxslt.x86_64 0:1.1.28-5.el7
nginxfilesystem.noarch 1:1.12.2-3.el7
nginx-mod-http-image-filter.x86_64 1:1.12.2-3.el7
nginx-mod-http-xslt-filter.x86_64 1:1.12.2-3.el7
nginx-mod-stream.x86_64 1:1.12.2-3.el7
dejavu-sans-fonts.noarch 0:2.33-6.el7
fontpackages-filesystem.noarch 0:1.44-8.el7
gperftools-libs.x86_64 0:2.6.1-1.el7
libX11-common.noarch 0:1.6.5-2.el7
libXpm.x86_64 0:3.5.12-1.el7
libxcb.x86_64 0:1.13-1.el7
nginx-all-modules.noarch 1:1.12.2-3.el7
nginx-mod-http-geoip.x86_64 1:1.12.2-3.el7
nginx-mod-http-perl.x86_64 1:1.12.2-3.el7
nginx-mod-mail.x86_64 1:1.12.2-3.el7

Complete!
```

2. 输入 `systemctl start nginx.service` 启动 Nginx 服务

说明

此处以 CentOS 7.3 64bit 为例，其他操作系统的 Nginx 启动命令请您自行查阅。

3. 输入 `wget http://127.0.0.1`，测试 Nginx 服务。

```

[root@bms-~]# wget http://127.0.0.1
--2019-07-04 11:06:32-- http://127.0.0.1/
Connecting to 127.0.0.1:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 3700 (3.6K) [text/html]
Saving to: 'index.html'

100%[=====>] 3,700 --.-K/s in 0s

2019-07-04 11:06:32 (532 MB/s) - 'index.html' saved [3700/3700]
    
```

访问Web默认页面

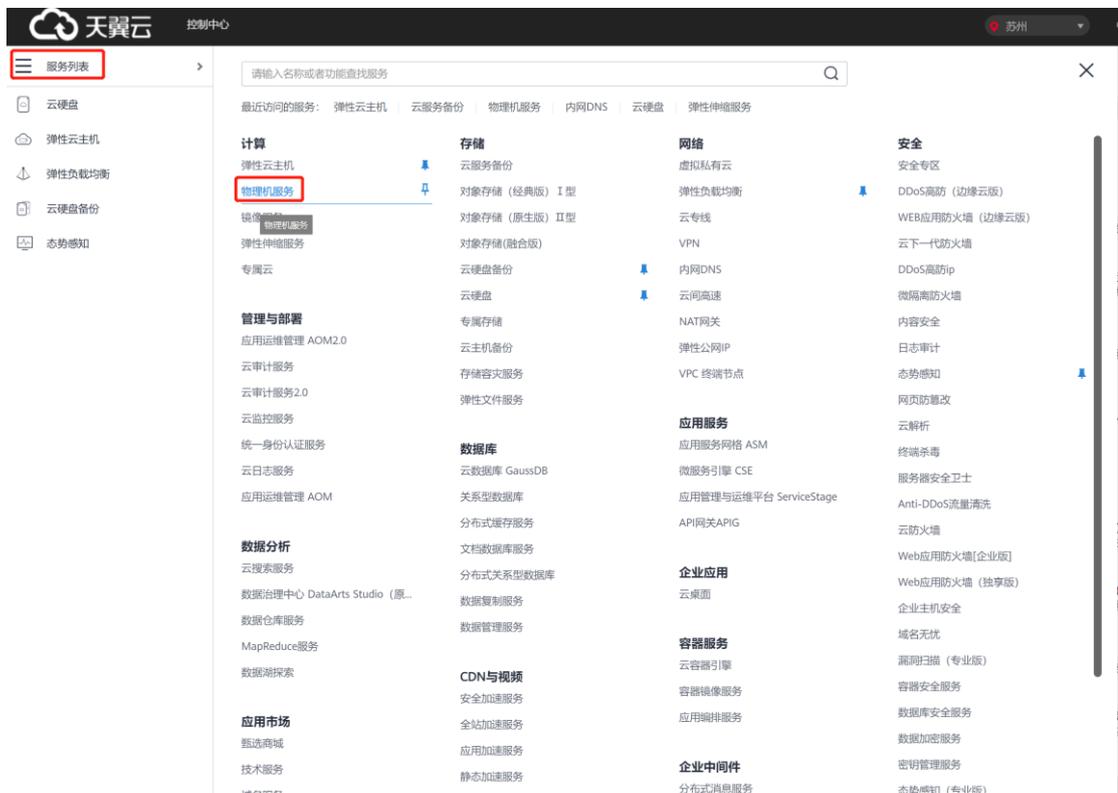
使用浏览器访问“http://物理机的弹性公网 IP 地址”，看到 Nginx 的欢迎页面即表示安装成功。

3.7 退订物理机服务

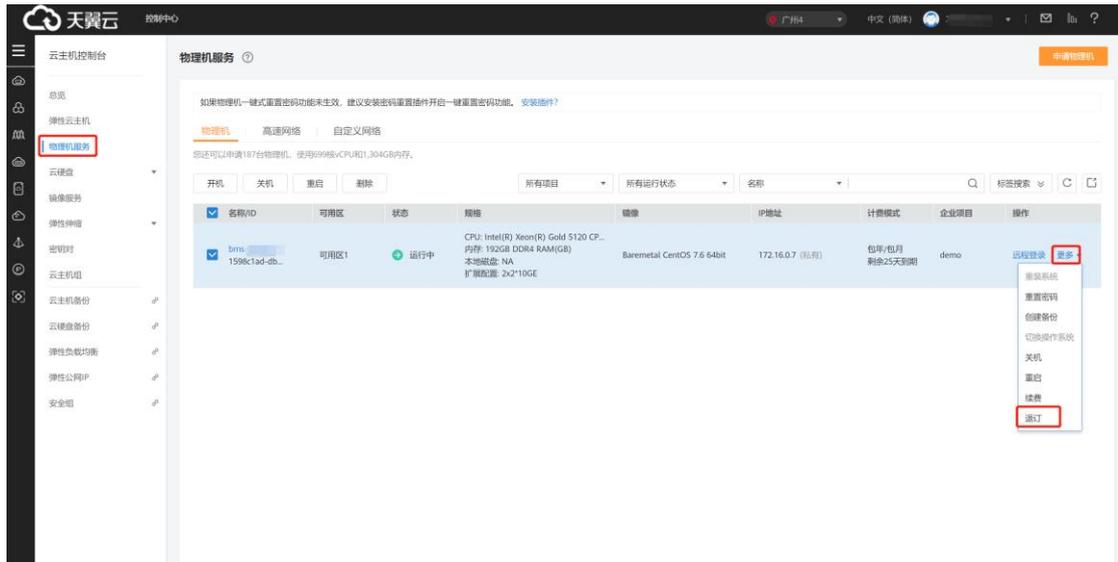
本章节主要介绍如何退订天翼云物理机。

操作步骤

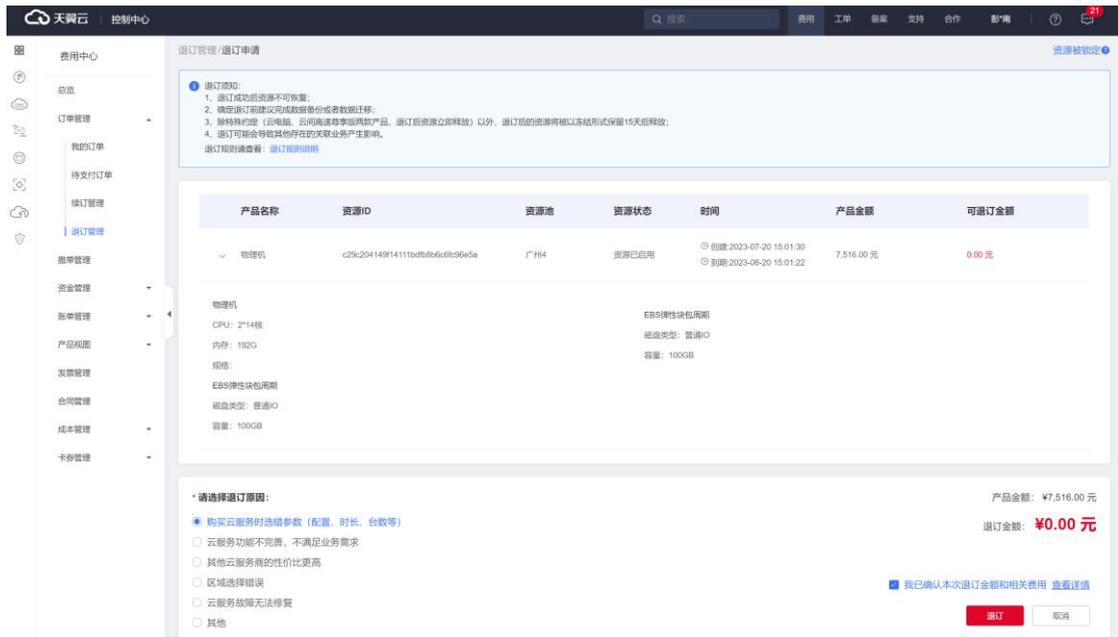
1. 登录天翼云服务控制中心。
2. 单击控制中心的区域按钮，选择区域和项目。
3. 在左侧导航栏，选择【计算】-【物理机服务】。



4. 在物理机列表中，找到已创建的实例“bms-xxx”。在“操作”列，单击“更多 > 退订”。



5. 在“退订资源”页面，选择一个退订原因，单击“退订”。



6. 在弹出的确认页面单击“退订”。

4 用户指南

4.1 物理机实例管理

4.1.1 物理机生命周期管理

物理机的生命周期管理包括开机、关机、重启和删除。

在云主机控制台导航栏中，选择【物理机服务】，进入物理机列表页，如下图所示。

通过以下两种方式，可以执行物理机的开机、关机、重启和删除操作：

1. 在物理机列表，通过点击该物理机所在列【操作】列下对应的按钮即可执行对应的操作。
2. 单击需要执行作的物理机右侧【更多】按钮，单击展开的【关机】/【重启】/【删除】按钮即可；

说明：

包年/包月物理机不能直接删除，需要进行退订。

名称/ID	状态	规格	镜像	私有IP地址	弹性IP	可用分区	操作
1606f241-bf95-4642-a9f...	运行中	CPU: Intel(R) Xeon(R) CP... 内存: 128GB DDR4 RAM... 本地硬盘: SAS 2*600G RA... 扩展配置: 2*10GE	Baremetal OracleLinux 7...	...	--	可用区	远程登录 更多
068f5d59-2cbe-4813-9b...	运行中	CPU: Intel(R) Xeon(R) CP... 内存: 128GB DDR4 RAM... 本地硬盘: SAS 2*600G RA... 扩展配置: 2*10GE	Baremetal OracleLinux 7...	...	--	可用区	远程登录 更多

物理机的所有状态如下表所示，根据提示信息，确认状态。

状态	状态属性	说明
申请中	中间状态	申请物理机后，物理机状态进入运行中之前的状态。 如果长时间处于该状态，则说明出现异常，需要联系管理员处理。
开机中	中间状态	物理机从关机到运行中的中间状态。 如果长时间处于该状态，则说明出现异常，需要联系管理员处理。
运行中	稳定状态	物理机正常运行中的状态。 在这个状态的服务器可以运行您的业务。
关机中	中间状态	物理机从运行中到关机的中间状态。 如果长时间处于该状态，则说明出现异常，需要联系管理员处理。
关机	稳定状态	物理机被正常停止。 在这个状态下的服务器，不能对外提供业务。
重启中	中间状态	物理机正在进行重启操作。 如果长时间处于该状态，则说明出现异常，需要联系管理员处理。
删除中	中间状态	物理机处于正在被删除的状态。 如果长时间处于该状态，则说明出现异常，需要联系管理员处理。
已删除	中间状态	物理机已被正常删除。在该状态下的服务器，不能对外提供业务，并在短时间内从系统中彻底清除。
故障	稳定状态	物理机处于异常状态。 在这个状态下的服务器，不能对外提供业务，需要联系管理员进行处理。

4.1.2 创建物理机

4.1.2.1 创建物理机服务器

本章节主要介绍如何创建天翼云物理机服务器。

4.1.2.1.1 操作场景

若您需要将您的应用服务部署在物理机服务器上，首先需要创建物理机服务器。

4.1.2.1.2 前提条件

- 在创建物理机服务器前，您已经完成[准备工作](#)。

4.1.2.1.3 操作步骤

- 登录管理控制台。
- 选择“计算 > 物理机服务器”。
进入物理机服务器页面。
- 单击“申请物理机服务器”，开始申请物理机服务器。
- 选择“区域”。
不同区域的云服务产品之间内网互不相通。建议您选择最靠近您业务的区域，这样可以减少网络时延、提高访问速度。需要注意：资源创建成功后不能更换区域。
- 选择“可用区”。
可用区指在同一区域下，电力、网络隔离的物理区域，可用区之间内网互通，不同可用区之间物理隔离。
 - 如果您需要提高应用的高可用性，建议您将物理机服务器创建在不同的可用区。
 - 如果您需要较低的网络时延，建议您将物理机服务器创建在相同的可用区。
- 选择“规格”。
包括规格名称、CPU、内存、本地磁盘和扩展配置。当您选择规格列表中的一个规格后，列表下方会展示该规格的名称、用途等信息，以便您根据业务场景进行选择。
其中，扩展配置描述了所选机型的网卡信息。例如：2 x 2*10GE 表示 1 块双网口的 10GE 网卡接入 VPC 网络，1 块双网口的 10GE 扩展网卡支持物理机服务器间的高速互联。

📖 说明

- 规格中的 CPU、内存、本地磁盘等配置为固定值，不可更改。
- 不同规格的物理机服务器带宽能力不同，请您根据实际业务慎重选择。
- 某些规格的物理机服务器支持快速发放能力，选择后，界面会提供“系统盘”参数（在“磁盘”配置项中展示），物理机服务器操作系统直接安装在云硬盘，实现快速发放。

7. 选择“镜像”。

- 公共镜像

常见的标准操作系统镜像，所有用户可见，包括操作系统以及预装的公共应用（SDI 卡驱动、bms-network-config 网络配置程序、Cloud-init 初始化工具等）。请根据您的实际需要自助配置应用环境或相关软件。

- 私有镜像

用户基于外部镜像文件或物理机服务器创建的个人镜像，仅用户自己可见。包含操作系统、预装的公共应用以及用户的私有应用。选择私有镜像创建物理机服务器，可以节省您重复配置物理机服务器的时间。

- 共享镜像

您将接受公有云其他用户共享的私有镜像，作为自己的镜像进行使用。

8. 选择“许可类型”。

在云平台上使用操作系统或软件的许可证类型。如果您选择的镜像为免费的，则系统不会展示该参数。如果您选择的镜像为计费镜像，此时系统会展示该参数。

- 使用平台许可证

使用云平台提供的许可证，申请许可证需要支付一定的费用。

- 使用自带许可证（BYOL）

使用用户已有操作系统的许可证，无需重新申请。

9. 设置“磁盘”。

根据磁盘使用的存储资源是否独享，磁盘划分为“云硬盘”、“专属存储”。其中，专属存储是为您提供的独享存储资源。

- 如果您在专属存储服务页面申请了存储池，可以选择“专属存储”页签，在已申请的存储池中创建磁盘。

- 如果未申请独享的存储池，请选择“云硬盘”页签，创建的磁盘使用公共存储资源。

📖 说明

- 使用专属存储资源创建磁盘时，待创建磁盘的磁盘类型需和申请的存储池资源类型保持一致，如都是“高 IO”类型。

磁盘包括系统盘和数据盘。您可以为物理机服务器添加多块数据盘，系统盘大小可以根据需要自定义。

- 系统盘

如果选择支持快速发放的规格，界面会提供系统盘配置项，可以根据需要设置磁盘类型和大小。

- 数据盘

您可以为物理机服务器添加多块数据盘，并设置数据盘的共享功能。

- 物理机服务器仅支持 SCSI 磁盘模式。
- 共享盘：勾选后，数据盘为共享云硬盘。该共享盘可以同时挂载给多台物理机服务器使用。

10. 设置网络信息，包括“虚拟私有云”、“网卡”、“安全组”等信息。

第一次使用云服务时，系统将自动为您创建一个默认的虚拟私有云，包括安全组、网卡。其中，默认虚拟私有云支持的地址范围为 192.168.1.0/24，子网网关为 192.168.1.1，并为子网开启 DHCP 功能。

4.1.2.1.4 网络参数说明

参数	解释
虚拟私有云	您可以选择使用已有的虚拟私有云，或者单击“新建虚拟私有云”创建新的虚拟私有云。
网卡	<p>包括主网卡和扩展网卡。您可以添加一张扩展网卡，并指定网卡（包括主网卡）的 IP 地址。</p> <p>须知</p> <ul style="list-style-type: none"> 主网卡用于系统的默认路由，不允许删除。 如果您选择自动分配 IP 地址，请不要在物理机服务器发放完成后修改私有 IP 地址，避免和其他物理机服务器 IP 冲突。 为网卡分配固定 IP 地址后，不能批量创建物理机服务器。
高速网卡	<p>物理机服务器之间的高速互连网络接口，为物理机服务器提供更高的带宽。</p> <p>同一台物理机服务器的多张高速网卡不能选择同一个高速网络。</p>
安全组	<p>安全组用来实现安全组内和安全组间物理机服务器的访问控制，加强物理机服务器的安全保护。用户可以在安全组中定义各种访问规则，当物理机服务器加入该安全组后，即受到这些访问规则的保护。</p> <p>创建物理机服务器时，仅支持选择一个安全组。但是物理机服务器创建成功后，可以为物理机服务器关联多个安全组。</p> <p>说明</p> <p>物理机服务器初始化需要确保安全组出方向规则至少满足如下要求：</p> <ul style="list-style-type: none"> 协议：TCP 端口范围：80 远端地址：169.254.0.0/16 <p>如果您使用的是默认安全组出方向规则，则已经包括了如上要求，可以正常初始化。默认安全组出方向规则为：</p> <ul style="list-style-type: none"> 协议：Any 端口范围：Any 远端地址：0.0.0.0/16
弹性公网 IP	<p>弹性公网 IP 是指将公网 IP 地址和路由网络中关联的物理机服务器绑定，以实现虚拟私有云内的物理机服务器通过固定的公网 IP 地址对外提供访问服务。</p> <p>您可以根据实际情况选择以下三种方式：</p>

参数	解释
	<ul style="list-style-type: none"> 不使用：物理机服务器不能与外部网络互通，仅可以在私有网络中部署业务或构建集群。 自动分配：自动为物理机服务器分配独享带宽的弹性公网 IP，带宽值由您设定。 使用已有：为物理机服务器分配已有弹性公网 IP。 <p>说明 选择已有弹性公网 IP 后，不能批量创建物理机服务器。</p>
规格	弹性公网 IP 选择“自动分配”时，需配置该参数。 <ul style="list-style-type: none"> 全动态 BGP：可根据设定的寻路协议第一时间自动优化网络结构，以保持客户使用的网络持续稳定、高效。 静态 BGP：网络结构发生变化时，运营商无法实时自动调整网络设置以保障用户体验。
计费方式	弹性公网 IP 选择“自动分配”时，需配置该参数。 <ul style="list-style-type: none"> 按带宽计费：指定带宽上限，按使用时间计费，与使用的流量无关。 按流量计费：按照实际使用的流量来计费。
带宽	弹性公网 IP 选择“自动分配”时，需配置该参数。 带宽大小，单位 Mbit/s。

11. 设置物理机服务器登录方式。

“密钥对”方式创建的物理机服务器安全性更高，建议选择“密钥对”方式。如果您习惯使用“密码”方式，请增强密码的复杂度，如 4.1.2.1.5 所示，保证密码符合要求，防止被恶意攻击。

- 密钥对

指使用密钥对作为登录物理机服务器的鉴权方式。您可以选择使用已有的密钥，或者单击“新建密钥对”创建新的密钥。

说明

如果选择使用已有的密钥，请确保您已在本地获取该文件，否则，将影响您正常登录物理机服务器。

- 密码

指使用设置初始密码方式作为物理机服务器的鉴权方式，此时，您可以通过用户名密码方式登录物理机服务器。

Linux 操作系统时为 root 用户的初始密码，Windows 操作系统时为 Administrator 用户的初始密码。密码复杂度需满足 4.1.2.1.5 要求。

4.1.2.1.5 密码规则

参数	规则	样例
密码	<ul style="list-style-type: none"> • 密码长度范围为 8 到 26 位。 • 密码至少包含以下 4 种字符中的 3 种： <ul style="list-style-type: none"> - 大写字母 - 小写字母 - 数字 - 特殊字符，包括!@\$%^&_+=[]{};:./? • 密码不能包含用户名或用户名的逆序。 • Windows 系统的物理机服务器，不能包含用户名中超过两个连续字符的部分。 	Test12\$@

12. （可选）高级配置。

如需使用“高级配置”中的功能，请单击“现在配置”。否则，请单击“暂不配置”。

- 标签：可选配置，对物理机的标识。使用标签可以方便识别和管理您拥有的物理机资源。
- 委托

委托是由租户管理员在统一身份认证服务（Identity and Access Management, IAM）上创建的，可以为物理机服务器提供访问云服务的临时凭证。

如果您在 IAM 上创建了委托，可以通过单击下拉列表选择委托名称，获取相应权限。目前，使用委托的场景为主机监控，物理机监控。

13. 设置“物理机服务器名称”。

名称可自定义，但需符合命名规则：只能由中文字符、英文字母、数字及“_”、“-”、“.”组成。

一次创建多台物理机服务器，系统会自动按序增加后缀。例如：输入 bms，服务器名称为 bms-0001，bms-0002，……。再次创建多台服务器时，命名从上次最大值连续增加，例如：输入 bms，已有服务器 bms-0010，新创服务器名称为 bms-0011、bms-0012、……，命名达到 9999 时，从 0001 开始。

14. 设置您创建的物理机服务器数量，最多为 24 台。

设置完成后，您可以通过单击“价格计算器”，查询当前配置的费用。

说明

- 当配额充足时，一次最多可以创建 24 台物理机服务器；若配额不足 24 台，一次最多可创建数量为配额余量。
- 在设置网络信息时，若您选择自定义网卡或高速网卡的 IP 地址，或者选择已有弹性公网 IP，则一次只能创建一台物理机服务器。

15. 单击“立即申请”。

16. 在确认规格页面，单击“提交”。

大约需要 30 分钟，物理机服务器的状态会变为“运行中”。选择支持快速发放的规格，大约只需 5 分钟即可获得一台物理机服务器。

📖 说明

您可以随时查看物理机服务器的创建状态，具体操作请参见查看创建状态。

4.1.2.2 创建快速发放物理机服务器

本章节主要介绍如何快速发放天翼云物理机服务器。

4.1.2.2.1 操作场景

在天翼云物理机服务器发放过程中，普通物理机服务器的操作系统需要从云端下载、安装，下载过程会消耗较长时间。基于云硬盘的物理机服务器的操作系统直接安装在云硬盘（即系统盘使用云硬盘），所以可以实现快速发放能力。

相比普通物理机服务器，快速发放型服务器有如下优势：

- 从云硬盘启动，发放时间缩短至 5min 左右。
- 支持整机备份，数据更有保障。
- 支持故障重建，保证业务快速恢复。
- 支持镜像导出，可将现有物理机服务器的配置复用在其他机器，节省重复配置的时间。

在物理机服务器创建界面，选择支持快速发放的规格，设置系统盘类型和容量，并按照需求配置其他参数，即可获取一台快速发放型物理机服务器。

4.1.2.2.2 操作步骤

您可以按照[创建物理机服务器](#)的描述创建快速发放物理机服务器。

在选择配置时，需要注意以下几点：

- 规格：选择支持快速发放的规格，可以选择 `physical.s4.medium`、`physical.s4.large`、`physical.s4.xlarge`、`physical.s4.2xlarge` 和 `physical.s4.3xlarge`。
- 镜像：只支持部分公共镜像，请以界面显示为准。
- 磁盘：设置“系统盘”的磁盘类型和大小。

4.1.2.3 通过私有镜像创建物理机服务器

本章节主要介绍如何通过私有镜像创建天翼云物理机服务器。

4.1.2.3.1 操作场景

如果您要创建一台物理机服务器，与现有的某台服务器拥有完全相同的操作系统和应用软件，那么您可以使用该服务器创建私有镜像，并根据这个私有镜像创建服务器。采用此方法可以节省您重复配置服务器的时间，提高交付效率。

4.1.2.3.2 背景信息

您需要使用以下任意一种方式创建私有镜像：

- 通过物理机服务器创建私有镜像
- 通过外部镜像文件创建私有镜像

4.1.2.3.3 操作步骤

您可以按照[创建物理机服务器](#)中的操作指导创建物理机服务器。

在配置参数时，需要注意以下两点：

- 区域：必须选择私有镜像所在的区域。
- 规格：不同物理机服务器规格所支持的操作系统有所差异，请参考[物理机服务器类型与支持的操作系统版本](#)并根据私有镜像的操作系统来选择规格。
- 镜像：选择“私有镜像”或“共享镜像”，并在下拉列表中选择需要的镜像。
- 磁盘：如果所选规格支持快速发放，那么系统盘在设置时，建议将容量调大 2GB 以上。

4.1.3 查看物理机信息

本章节主要介绍如何查看天翼云物理机信息。

4.1.3.1 查看创建状态

本章节主要介绍如何查看天翼云物理机的创建状态。

4.1.3.1.1 操作场景

用户创建物理机服务器后，可以通过“申请状态”栏查看任务的创建状态。创建物理机服务器的任务可以包括创建物理机服务器资源、绑定弹性公网 IP、挂载云硬盘等子任务。

申请状态栏的任务状态包括如下两类：

- 处理中：指系统正在处理的请求。

- **处理失败：**指未能成功处理的请求。对于处理失败的任务，系统会自动回退，同时在界面上直观的展示错误码，例如“（BMS.3033）创建系统卷失败。”

本节介绍如何查看物理机服务器的申请状态以及申请状态栏的提示信息说明。

4.1.3.1.2 操作步骤

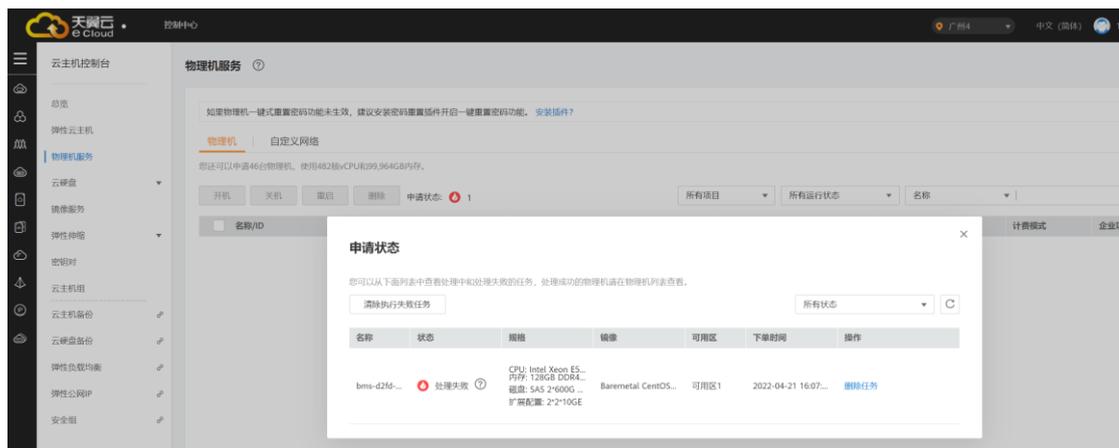
1. 登录管理控制台。
2. 选择“计算 > 物理机服务”。进入物理机服务器页面。
3. 常用操作“开机、关机、重启、删除”的右侧即为“申请状态”，用户执行创建物理机服务器操作后，申请状态栏将显示该任务的创建状态。

图 1 申请状态



4. 单击“申请状态”栏对应的数字，即可查看“处理中”的任务和“处理失败”任务的详情，查看物理机服务器的创建状态。

图 2 申请状态详情



说明

如果用户发现申请状态栏显示创建物理机服务器的任务失败，而物理机服务器列表中显示该服务器已创建成功，关于此问题请参见[为什么创建物理机服务器的任务失败，但是在服务器列表中显示创建成功](#)。

4.1.3.2 查看详细信息

本章节主要介绍如何查看天翼云物理机的详细信息。

4.1.3.2.1 操作场景

在您创建了物理机服务器后，可以通过管理控制台查看和管理您的物理机服务器。本节介绍如何查看物理机服务器的详细信息，包括物理机服务器名称/ID、磁盘、网卡、弹性公网 IP 等信息。

4.1.3.2.2 操作步骤

1. 登录管理控制台。
2. 选择“计算 > 物理机服务器”。
进入物理机服务器列表页面，您可以在本页面查看您已创建的物理机服务器，以及物理机服务器的规格、镜像、私有 IP 地址等基本信息。
3. 在物理机服务器列表中的右上方，按照运行状态、名称、物理机服务器 ID、规格和私有 IP 地址筛选需要的物理机服务器。
4. 单击物理机服务器的名称。
进入物理机服务器详情界面。
5. 查看物理机服务器的详细信息，如：名称、状态、规格、虚拟私有云等信息。还可以单击“磁盘/网卡/安全组/弹性公网 IP/监控”页签，为物理机服务器挂载、卸载云硬盘，更改安全组，绑定、解绑定弹性公网 IP，添加委托等。

📖 说明

物理机服务器监控数据与图表不在详情界面展示，您需要在云监控界面查看。当然，前提是您的物理机服务器已安装主机监控 Agent，详情请参见安装配置 Agent。

4.1.3.3 导出物理机服务列表

本章节主要介绍如何导出天翼云物理机列表信息。

操作场景：

您可以将当前帐号下拥有的所有物理机信息，以 CSV 文件的形式导出至本地。该文件以“物理机-区域-当前日期”命名，记录了物理机的名称、ID、可用分区、状态、规格等信息。

操作步骤：

1. 登录管理控制台。
2. 单击管理控制台左上角的 ，选择区域和项目。
3. 选择“计算 > 物理机”。

4. 在物理机列表页，单击右上角的 。

系统会将您帐号下，当前 Region 的所有物理机信息自动导出，并下载至本地。



4.1.4 重置物理机的密码

本章节主要介绍如何重置天翼云物理机的密码。

公有云平台提供了一键式重置密码功能，当物理机的密码丢失或过期时，如果您的物理机提前安装了密码重置插件，可以参见本节内容获取新密码。

前提条件:

- 密码丢失或过期前，已安装密码重置插件“Cloud-reset-pwd Agent”。安装操作请参见安装一键式重置密码插件（可选）。
- 物理机使用的 VPC 网络 DHCP 不能禁用。
- 物理机网络正常通行。
- 物理机已绑定弹性 IP。
- 物理机系统镜像不能关联至数据盘

操作步骤:

1. 登录管理控制台。
2. 单击管理控制台左上角的 ，选择地域和项目。
3. 选择“计算 > 物理机服务”。

进入物理机页面。

4. 选中待重置密码的物理机，并选择“操作”列下的“更多 > 重置密码”。
5. 根据界面提示，设置物理机的新密码，并确认新密码。密码规则如下表。

参数	规则	样例
----	----	----

密码	密码长度范围为 8 到 26 位。 密码至少包含以下 4 种字符中的 3 种： 大写字母 小写字母数字 特殊字符 Windows: ~!@#%*- _=+[]:/? Linux: ~!@# % ^*_-=+[]{};.,/? 密码不能包含用户名或用户名的逆序。 Windows 系统的物理机，不能包含用户名中超过两个连续字符的部分。	Test12@#
----	--	----------

6. 单击“确定并重启”。

系统执行重置密码操作，该操作预计需要 10 分钟，请勿频繁执行。重置过程中，系统会自动重启物理机。

4.1.5 重装物理机操作系统

本章节主要介绍如何重装天物理机操作系统。

物理机操作系统无法正常启动，操作系统中毒，或物理机系统运行正常，但需要对系统进行优化，使其在最优状态下工作时，用户可以使用重装物理机的操作系统功能。

重装须知：

- 支持对本地磁盘发放和快速发放物理机的系统进行重装。
- 重装过程中禁止对物理机进行关机或重启等操作。
- 重装操作系统成功后，物理机会自动开机。

约束与限制：

仅支持同一个操作系统重装。

- 如果物理机的启动设备为云硬盘，需确保云硬盘的配额大于 0。
- 如果是通过私有镜像创建的物理机，请确保原有镜像仍存在。
- 重装系统界面不支持本地磁盘物理机的系统盘容量显示。
- 重装过程不支持更改许可证类型，也不支持用户数据注入。

前提条件：

- 待重装操作系统的物理机需处于“关机”状态。

- 待重装操作系统的快速发放物理机挂载有系统盘。
- 重装操作系统会清除系统盘数据，包括系统盘上的系统分区和所有其他分区，请做好数据备份。

操作步骤:

1. 登录管理控制台。
2. 单击管理控制台左上角的 ，选择地域和项目。
3. 选择“计算 > 物理机服务”，进入物理机页面。
4. 在待重装操作系统的物理机的“操作”列下，单击“更多 > 重装系统”，进入“重装系统”界面。
5. 设置登录方式。

密钥对：您可以选择已有密钥对，或单击“查看密钥对”新建一个密钥对作为登录物理机的私钥。

密码：您可以设置操作系统的初始登录密码。密码必须符合密码复杂度要求，符合密码规则。

说明：

Windows 操作系统只支持设置密钥登录。

6. 单击“确定”。

在“物理机重装系统”页面，确认重装的操作系统规格无误后，单击“提交申请”。

7. 提交重装系统的申请后，物理机的状态变为“重装中”，当该状态消失后，表示重装结束。

说明：在重装操作系统过程中请不要对该物理机进行任何操作。

8. （可选）如果物理机在重装操作系统前已配置 QinQ 网络或 IB 网络，重装完成后请参考 [自定义 vlan 网络](#)。

4.1.6 备份物理机

本章节主要介绍物理机的备份。

操作场景

为了保证数据安全，您可以对物理机中的所有云硬盘（系统盘和数据盘）进行备份，采用这种备份方式可以避免因备份创建时间差带来的数据不一致问题。云服务器备份支持基于多云硬盘一致性快照技术的备份服务，并支持利用备份数据恢复物理机数据，最大限度保障用户

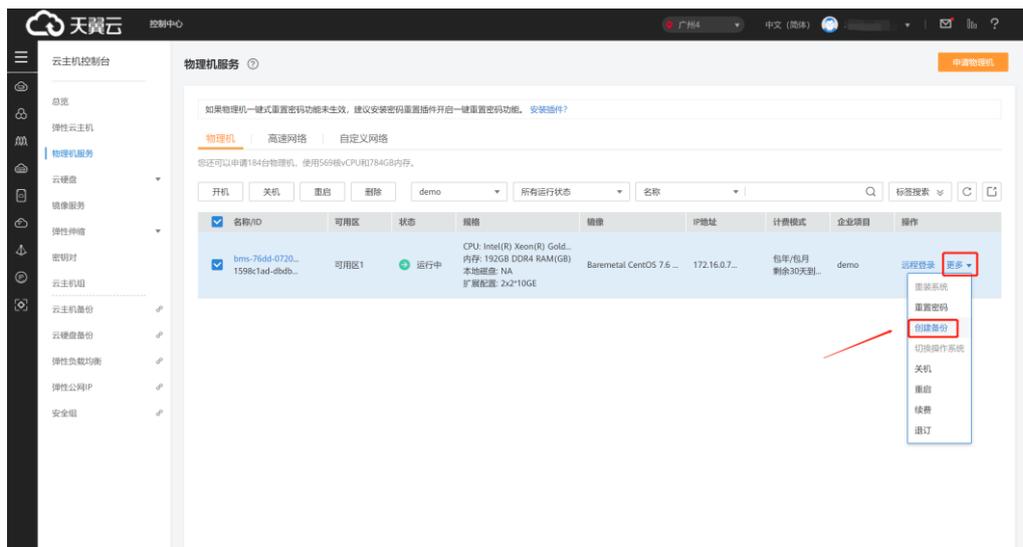
数据的安全性和正确性，确保业务安全。

约束与限制

- 不支持使用物理机的备份创建镜像。
- 不支持备份带有共享云硬盘的物理机。
- 恢复时，物理机会自动关机，将中断租户业务，关机后有一段时间物理机处于锁定状态，租户不可操作。

操作步骤

1. 登录管理控制台。
2. 选择“计算 > 物理机服务”。
进入物理机列表页面。
3. 在待备份的物理机所在行，单击“更多 > 创建备份”。
系统跳转至“创建云服务器备份”页面。



4. 根据界面提示完成如下操作：
 - 选择服务器：在服务器列表中，默认会勾选待创建备份的物理机，保持默认即可。
 - 备份配置：勾选“自动备份”，选择一个备份策略。

创建云主机备份
[返回云主机备份列表](#)

区域 广州4 如需切换区域, 请单击页面上角的下拉列表。
不同区域的资源之间内网不互通, 请选择靠近您客户的区域, 可以降低网络时延, 提高访问速度。

选择服务器

服务器列表 所有状态 名称 bms-... x Q C

<input checked="" type="checkbox"/>	名称/ID	状态	类型	可用区	绑定状态
<input checked="" type="checkbox"/>	bms-76dd... 1598c1ad-...	运行中	物理机	可用区1	是 (backup_...

已勾选服务器列表 (1) 名称 Q

<input checked="" type="checkbox"/>	名称/ID	状态	类型	可用区	绑定状态	操作
<input checked="" type="checkbox"/>	bms-7... 1598c...	运行中	物理机	可用区1	是 (backu...	

备份配置

自动备份
将选择的服务器绑定到备份策略中, 按照备份策略进行自动备份。如果选择的服务器已绑定其他策略, 绑定操作会使它从原策略解绑并绑定到所选策略。

* 备份策略 backup_policy_188338 | 启用 | 00:30 | 星期一 | 30天 C 创建策略

立即备份

说明

将选择的物理机绑定到备份策略中, 按照备份策略进行自动备份。

如果选择的物理机已绑定其他策略, 在选择新的备份策略后, 服务器会自动从原备份策略解绑, 并绑定到新的备份策略。

您也可以勾选“立即备份”, 选择的物理机将立即进行一次备份。

输入备份名称和描述, 如果是首次备份, 建议启动全量备份。

备份配置

自动备份
将选择的服务器绑定到备份策略中, 按照备份策略进行自动备份。如果选择的服务器已绑定其他策略, 绑定操作会使它从原策略解绑并绑定到所选策略。

* 备份策略 backup_policy_188338 | 启用 | 00:30 | 星期一 | 30天 C 创建策略

立即备份
选择的服务器将立即进行一次备份。

* 名称 ?

描述 0/255

全量备份 启用 ?

数据库服务器备份 启用 ?

配置费用 ¥0.27/GB/月
点击查看 查看新买硬盘包的详细价格

立即申请

4.2 制作物理机镜像

4.2.1 私有镜像概述

本章节主要描述私有镜像。

私有镜像为用户自己创建的镜像，包含操作系统或业务数据、预装的公共应用以及用户的私有应用，仅用户个人可见。您可以通过以下方式创建私有镜像：

- 通过物理机服务器创建私有镜像

📖 说明

当前仅支持快速发放型物理机服务器（操作系统安装在云硬盘中）来创建私有镜像。

- 通过外部镜像文件创建私有镜像

用户可以上传外部镜像文件并将镜像注册到云平台上，成为自己的私有镜像。目前支持的外部镜像文件格式包括：VMDK、VHD、QCOW2、RAW、VHDX、QED、VDI、QCOW、ZVHD2 和 ZVHD。

📖 说明

其他格式的镜像文件需要预先使用工具转换镜像格式后，才能运行在物理机服务器。

当您成功创建私有镜像后，镜像的状态为“正常”。此时，您可以使用该镜像新建物理机服务器实例，也可以将其共享给其他用户，还可以复制到您帐号下的其他区域。

4.2.2 通过物理机服务器创建私有镜像

本章节主要介绍如何通过物理机服务器创建私有镜像。

4.2.2.1 操作场景

您可以基于物理机服务器实例创建私有镜像，将实例的系统盘数据完整地复制到私有镜像中。系统盘一般包含用户运行业务所需的操作系统、应用软件。

4.2.2.2 约束限制

- 只有系统盘为云硬盘时，才支持此操作。
- 暂不支持将物理机服务器实例的数据盘导出为镜像。
- 物理机服务器实例必须为“关机”状态。
- 此操作依赖于物理机服务器镜像中的 `bms-network-config` 和 `Cloud-Init` 插件。

- 如果待创建私有镜像的物理机服务器使用的是公共镜像，镜像中已内置 bms-network-config 和 Cloud-Init 插件。
- 如果待创建私有镜像的物理机服务器使用的是私有镜像，请确认是否已安装并配置 bms-network-config 和 Cloud-Init 插件。

4.2.2.3 操作须知

- 请将物理机服务器实例中的敏感数据删除后再创建私有镜像，避免数据安全隐患。
- 请将操作系统中的残留文件进行清理。请参考临时文件清理
- 创建私有镜像的过程中，请不要改变实例的状态，避免创建失败。

4.2.2.4 操作步骤

1. 登录管理控制台。
2. 选择“计算 > 物理机服务”。
进入物理机服务器页面。
3. 在需要创建私有镜像的物理机服务器的“操作”列，单击“更多 > 关机”。
只有关机状态的物理机服务器才能制作私有镜像。
4. 待物理机服务器状态变为“关机”时，单击“操作”列的“更多 > 制作镜像”。
进入“创建镜像”页面。
5. 填写镜像名称，根据需要设置标签并输入该镜像的描述。
设置完成后，单击“立即申请”。
6. 在“确认规格”页面，确认规格无误后，单击“提交”。
7. 页面跳转至镜像列表，可以看到正在创建的私有镜像，待状态变为“正常”时，表示创建成功。

4.2.2.5 后续操作

成功创建私有镜像后，您可能想[通过私有镜像创建物理机服务器](#)。在创建物理机服务器页面，选择私有镜像下拉框中的镜像。

4.2.2.6 附：临时文件清理

- 执行下面命令，清理用户登录记录。

```
echo > /var/log/wtmp
```

```
echo > /var/log/btmp
```

- 执行下面命令，清理相应目录下的临时文件。

```
rm -rf /var/log/cloud-init*
```

```
rm -rf /var/lib/cloud/*
```

```
rm -rf /var/log/network-config.log
```

- 执行下面命令，清理残留配置信息。
 - SUSE 操作系统：查看“/etc/sysconfig/network/”文件夹下有哪些以“ifcfg”开头的文件，删除除了“ifcfg-lo 及 ifcfg.template”以外的以“ifcfg”开头的文件。
查看文件命令：`ll /etc/sysconfig/network/`
删除文件命令：`rm -rf /etc/sysconfig/network/ifcfgxxx`
 - RedHat/CentOS/Oracle/Euler 操作系统：查看“/etc/sysconfig/network-scripts/”文件夹下有哪些以“ifcfg”开头的文件，删除除了“ifcfg-lo”以外的以“ifcfg”开头的文件。
查看文件命令：`ll /etc/sysconfig/network-scripts/`
删除文件命令：`rm -rf /etc/sysconfig/network-scripts/ifcfgxxx`
 - Ubuntu 操作系统：`rm -rf /etc/network/interfaces.d/50-cloud-init.cfg`
- 执行下面命令清除历史操作记录。
`history -w;echo > /root/.bash_history;history -c;history -c;history -c;`

4.2.3 通过外部镜像文件创建私有镜像

本章节主要介绍如何通过外部镜像文件创建私有镜像。

4.2.3.1 操作场景

您可以通过外部镜像文件创建私有镜像并根据私有镜像创建物理机实例。

4.2.3.2 制作方式

提前准备外部镜像文件，提交工单，联系天翼云运维团队协助您快速完成私有镜像的制作。

4.3 存储

4.3.1 挂载磁盘

本章节主要介绍如何给物理机挂载磁盘。

物理机器创建成功后，如果发现磁盘不够用或当前磁盘不满足要求，可以将已有磁盘挂载给物理机器，或在“存储”页面创建新的磁盘，然后再挂载至物理机器。

前提条件：

已创建可用的磁盘。

创建云硬盘的操作，请参见《云硬盘用户指南》的“[创建云硬盘](#)”章节。

1. 登录管理控制台。
2. 单击管理控制台的区域按钮，选择地域和项目。
3. 选择“计算 > 物理机服务”。
4. 在物理机器列表中的右上角，输入物理机器名称、IP 地址或 ID，进行搜索。
5. 单击待挂载磁盘的物理机器的名称。
6. 系统跳转至该物理机器详情页面，选择“磁盘”页签，并单击“挂载磁盘”。
7. 系统跳转至“挂载磁盘”页面，根据界面提示选择磁盘类型，勾选目标磁盘，并设置挂载点。

说明：

4. 如果无可用的云硬盘，请单击列表下方的“创建云硬盘”进行创建。

挂载磁盘时的约束限制，请参见[物理机挂载磁盘时有什么限制？](#)。

8. 单击“确定”。
9. 挂载成功后，在物理机器详情页的“磁盘”页签，即可看到新挂载的磁盘信息。
10. 后续处理

如果挂载的云硬盘是新创建的，则云硬盘挂载至物理机器后，需要登录物理机器初始化云硬盘（即格式化云硬盘），之后云硬盘才可以正常使用。初始化数据盘的具体操作请参见[初始化数据盘](#)

说明：

物理机器重启后，云硬盘盘符可能发生变化，请参考[如何查看云硬盘盘符？](#)获取云硬盘设备和盘符的对应关系。

4.3.2 卸载磁盘

本章节主要介绍如何给物理机卸载磁盘。

卸载磁盘

将挂载至物理机器中的磁盘卸载。

对于挂载在系统盘盘位（也就是“/dev/sda”挂载点）上的磁盘，当前仅支持离线卸载。即卸载磁盘时，物理机器需处于“关机”状态。

对于挂载在数据盘盘位（非“/dev/sda”挂载点）上的磁盘，不仅支持离线卸载，还支持在线卸载磁盘功能。此时，物理机器处于“运行中”状态。

前提条件：

- 对于 Windows 物理机器，在线卸载磁盘前，请确保没有程序正在对该磁盘进行读写操作。否则，将造成数据丢失。
- 对于 Linux 物理机器，在线卸载磁盘前，客户需要先登录物理机器，执行 `umount` 命令，取消待卸载磁盘与文件系统之间的关联，并确保没有程序正在对该磁盘进行读写操作。否则，卸载磁盘将失败。

操作步骤：

在物理机器页面，单击待卸载磁盘的物理机器名称，跳转至物理机器详情页。

选择“磁盘”页签，单击待卸载磁盘所在行的“卸载”，卸载磁盘。

4.3.3 扩容磁盘

本章节主要介绍如何给物理机扩容磁盘。

当云硬盘空间不足时，可以为其扩容，系统盘和数据盘均支持扩容，系统盘的大小

上限为1TB。扩容方法请参考《云硬盘用户指南》中的“[扩容云硬盘](#)”章节。

扩容成功后，还需要对扩容部分的云硬盘分配分区：

对系统盘的扩容后处理请参见《云硬盘用户指南》中“Linux云硬盘扩容后处理

(fdisk)”章节的“替换原有分区”部分。

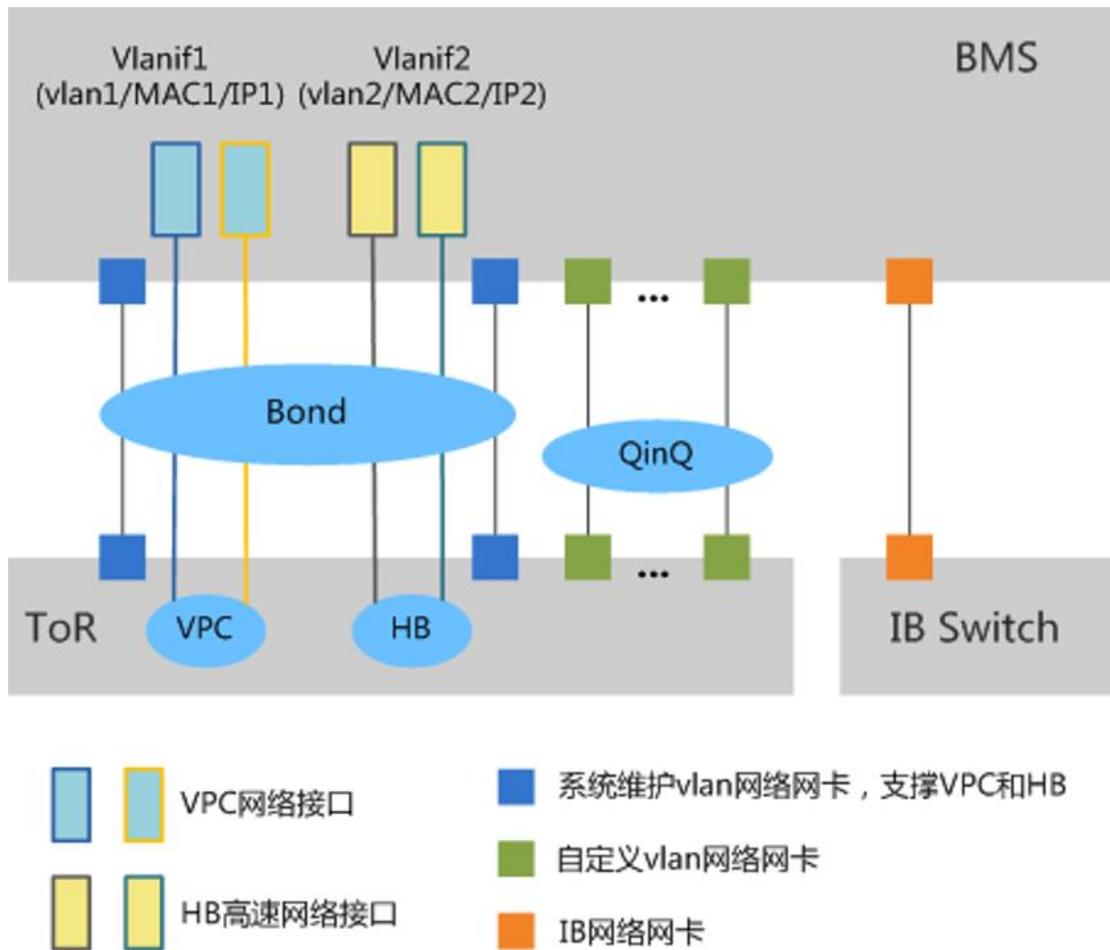
4.4 网络

4.4.1 物理机网络介绍

本章节主要介绍天翼云物理机的物理网络。

物理机有四种网络类型，分别是 VPC 网络、高速网络、自定义 vlan 网络和 IB 网络，四种网络之间相互隔离不互通。其中，VPC 网络接口和高速网络接口是基于系统维护 vlan 网卡组建 bond 后创建的 vlan 子接口。自定义 vlan 网络网卡和 IB 网络网卡由用户进行管理配置。

物理机网络：



说明：

- 图中的 ToR 表示服务器机柜的布线方式，接入交换机放在机架顶部，服务器放在下方。HB 表示高速网络。QinQ 表示 802.1Q 隧道。
- VPC 网络接口和 HB 高速网络接口由系统生成，租户不可修改。这两个网络接口属于同一个网卡 Bond。

- 弹性云服务器和物理机之间可以通过 VPC 网络通信，只有 VPC 网络支持安全组、弹性公网 IP 和弹性负载均衡能力。
- 对于高速网络和自定义 vlan 网络，同一网络中的物理机实例之间仅提供 2 层连接。

4.4.2 VPC 网络

本章节主要介绍天翼云物理机的 VPC 网络。

VPC 网络为物理机构建一个逻辑上完全隔离的专有区域。使用 1G port（千兆网络）。用户可以在 VPC 网络中配置弹性公网 IP、安全组和 VPN 等网络特性，并且该网络可用于弹性云服务器和物理机之间通信。通常物理机的 vpc 网络主网卡为 eth0 和 eth1 组成的 bond0（如 vpc 网络创建多块网卡，则网卡在物理机内是 bond0 的子 bond）。

查看方式

VPC 网络的网络接口可以在管理控制台中查看，也可以根据分配的 IP 地址在 Linux 系统中找到对应的 bond 接口。

磁盘 | **网卡** | 安全组 | 弹性IP | 监控 | 标签

192.168.0.21		修改私有IP 管理虚拟IP地址 更改安全组	
网卡ID	d45b22c4-44d9-4b76-8e3c-361a0a320ed9	状态	● 激活
弹性IP	--	子网	dualstacksubnet (192.168.0.0/24)
安全组	sg-ipv6	私有IP地址	192.168.0.21
源/目的检查	<input checked="" type="checkbox"/> ①	虚拟IP地址	--
VLAN	3709	MAC地址	fa:16:3eef:47:5c
192.168.1.90		管理虚拟IP地址 更改安全组	
网卡ID	4f4e987d-dfc5-4427-a8ef-74222a2d340c	状态	● 激活
弹性IP	--	子网	subnet-6ba8 (192.168.1.0/24)
安全组	sg-ipv6	私有IP地址	192.168.1.90
源/目的检查	<input checked="" type="checkbox"/> ①	虚拟IP地址	--
VLAN	3710	MAC地址	fa:16:3eef:48:a1
10.10.0.10 高速网卡			
10.10.1.173 高速网卡			

```
[root@server15162dc9-07e2-4daf-9611-2b5b02f60f0f ~]# ifconfig -a
bond0: flags=5187<UP,BROADCAST,RUNNING,MASTER,MULTICAST> mtu 8888
    inet 192.168.0.21 netmask 255.255.255.0 broadcast 192.168.0.255
    inet6 fe80::f816:3eff:feef:475c prefixlen 64 scopeid 0x20<link>
    ether fa:16:3e:ef:47:5c txqueuelen 1000 (Ethernet)
    RX packets 146 bytes 16818 (16.4 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 236 bytes 37932 (37.0 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

bond0.3710: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 8888
    inet 192.168.1.90 netmask 255.255.255.0 broadcast 192.168.1.255
    inet6 fe80::f816:3eff:feef:48a1 prefixlen 64 scopeid 0x20<link>
    ether fa:16:3e:ef:48:a1 txqueuelen 1000 (Ethernet)
    RX packets 3 bytes 741 (741.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 40 bytes 7611 (7.4 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

bond0.3806: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 8888
    inet 10.10.1.173 netmask 255.255.255.0 broadcast 10.10.1.255
    inet6 fe80::f816:3eff:fe8b:caac prefixlen 64 scopeid 0x20<link>
    ether fa:16:3e:8b:ca:ac txqueuelen 1000 (Ethernet)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 32 bytes 4198 (4.0 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

bond0.3807: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 8888
    inet 10.10.0.10 netmask 255.255.255.0 broadcast 10.10.0.255
    inet6 fe80::f816:3eff:febl:e505 prefixlen 64 scopeid 0x20<link>
    ether fa:16:3e:b1:e5:05 txqueuelen 1000 (Ethernet)
    RX packets 1 bytes 123 (123.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 44 bytes 5302 (5.1 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

eth0: flags=6211<UP,BROADCAST,RUNNING,SLAVE,MULTICAST> mtu 8888
    ether fa:16:3e:ef:47:5c txqueuelen 1000 (Ethernet)
    RX packets 131 bytes 14898 (14.5 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 259 bytes 42158 (41.1 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

eth1: flags=6211<UP,BROADCAST,RUNNING,SLAVE,MULTICAST> mtu 8888
    ether fa:16:3e:ef:47:5c txqueuelen 1000 (Ethernet)
    RX packets 33 bytes 3108 (3.0 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 72 bytes 8424 (8.2 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 72 bytes 8424 (8.2 KiB)
```

4.4.3 高速网络

本章节主要介绍天翼云物理机的高速网络。

高速网络是物理机的内部网络，与 VPC 共享一个物理平面，用户通过管理控制台创建高速网络后，系统会在物理机操作系统中建立一个专有的 vlan 子接口（通常为 bond0 的一个子 bond，名称为 bond0.***@bond0）。使用 10G port（万兆网络）。高速网络只有东西流量，且没有三层路由功能，用于节点间内部二层通信。

查看方式

高速网络的网络接口可以在管理控制台中查看，也可以根据分配的 IP 地址在 Linux 系统中查看。

磁盘 | **网卡** | 安全组 | 弹性IP | 监控 | 标签

192.168.0.21	修改私有IP	管理虚拟IP地址	更改安全组
192.168.1.90		管理虚拟IP地址	更改安全组
10.10.0.10 高速网卡			
网卡ID	0472d9f5-9dac-49a1-8c33-6a43e68042cc	状态	● 激活
VLAN	3807	私有IP地址	10.10.0.10
高速网络	highway-10086 (10.10.0.0/24)		
MAC地址	fa:16:3e:b1:e5:05		
10.10.1.173 高速网卡			
网卡ID	e7fc9900-6807-45e1-982e-4f9efc360229	状态	● 激活
VLAN	3806	私有IP地址	10.10.1.173
高速网络	highway-1008611 (10.10.1.0/24)		
MAC地址	fa:16:3e:8b:ca:ac		

```
[root@server15162dc0-07e2-4daf-9611-2b5b02f60f0f ~]# ifconfig -a
bond0: flags=5187<UP,BROADCAST,RUNNING,MASTER,MULTICAST> mtu 8888
    inet 192.168.0.21 netmask 255.255.255.0 broadcast 192.168.0.255
    inet6 fe80::f816:3eff:feef:475c prefixlen 64 scopeid 0x20<link>
    ether fa:16:3e:ef:47:5c txqueuelen 1000 (Ethernet)
    RX packets 146 bytes 16818 (16.4 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 236 bytes 37932 (37.0 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

bond0.3710: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 8888
    inet 192.168.1.90 netmask 255.255.255.0 broadcast 192.168.1.255
    inet6 fe80::f816:3eff:feef:48a1 prefixlen 64 scopeid 0x20<link>
    ether fa:16:3e:ef:48:a1 txqueuelen 1000 (Ethernet)
    RX packets 3 bytes 741 (741.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 40 bytes 7611 (7.4 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

bond0.3806: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 8888
    inet 10.10.1.173 netmask 255.255.255.0 broadcast 10.10.1.255
    inet6 fe80::f816:3eff:fe8b:caac prefixlen 64 scopeid 0x20<link>
    ether fa:16:3e:8b:ca:ac txqueuelen 1000 (Ethernet)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 32 bytes 4198 (4.0 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

bond0.3807: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 8888
    inet 10.10.0.10 netmask 255.255.255.0 broadcast 10.10.0.255
    inet6 fe80::f816:3eff:feb1:e505 prefixlen 64 scopeid 0x20<link>
    ether fa:16:3e:b1:e5:05 txqueuelen 1000 (Ethernet)
    RX packets 1 bytes 123 (123.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 44 bytes 5302 (5.1 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

eth0: flags=6211<UP,BROADCAST,RUNNING,SLAVE,MULTICAST> mtu 8888
    ether fa:16:3e:ef:47:5c txqueuelen 1000 (Ethernet)
    RX packets 131 bytes 14898 (14.5 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 259 bytes 42158 (41.1 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

eth1: flags=6211<UP,BROADCAST,RUNNING,SLAVE,MULTICAST> mtu 8888
    ether fa:16:3e:ef:47:5c txqueuelen 1000 (Ethernet)
    RX packets 33 bytes 3108 (3.0 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 72 bytes 8424 (8.2 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 72 bytes 8424 (8.2 KiB)
```

4.4.4 自定义 VLAN 网络

本章节主要介绍天翼云物理机的自定义 VLAN 网络。

介绍

未被系统默认使用的以太网卡，系统内显示为两个网口一组，常见命名为：enp2s0f1，enp129s0f1 或 eth3，cirename0，默认为 10GE，万兆网络）可用于自定义 VLAN 网络，物理上采用 QinQ 技术实现用户的网络隔离，提供额外的物理平面，通常用户通过配置 bond 实现高可用。同一用户名下的物理机默认处于同一 vlan 里，故用户设定同一网段 ip 即可通信，适用于 Oracle RAC、SAP HANA 和 VMware 等场景，如 Oracle RAC 适用于做心跳线。自定义 VLAN 网络当前不支持跨 AZ 互通。

系统默认未使用的以太网卡没有默认配置文件，在系统启动时处于 down 状态，可以使用 ip a 命令查看网卡名。例如，一台 Linux 操作系统的物理机，eth0/eth1 自动组 bond0 用于 VPC 网络；enp2s0f1、enp129s0f1 用于自定义 VLAN 网络。

如图：默认两个网口都是 down 的状态，需手动配置，如何配置可参考自定义网络配置。

```
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN qlen 1
   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,SLAVE,UP,LOWER_UP> mtu 8888 qdisc mq master bond0 state UP qlen 1000
   link/ether fa:16:3e:28:03:fe brd ff:ff:ff:ff:ff:ff
3: enp2s0f1: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN qlen 1000
   link/ether ac:75:1d:7b:e3:8f brd ff:ff:ff:ff:ff:ff
4: eth1: <BROADCAST,MULTICAST,SLAVE,UP,LOWER_UP> mtu 8888 qdisc mq master bond0 state UP qlen 1000
   link/ether fa:16:3e:28:03:fe brd ff:ff:ff:ff:ff:ff
5: enp129s0f1: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN qlen 1000
   link/ether 8c:6d:77:f8:d3:9f brd ff:ff:ff:ff:ff:ff
6: bond0: <BROADCAST,MULTICAST,MASTER,UP,LOWER_UP> mtu 8888 qdisc noqueue state UP qlen 1000
   link/ether fa:16:3e:28:03:fe brd ff:ff:ff:ff:ff:ff
   inet 192.168.111.67/24 brd 192.168.111.255 scope global dynamic bond0
       valid_lft 86206sec preferred_lft 86206sec
   inet6 fe80::f816:3eff:fe28:3fe/64 scope link
       valid_lft forever preferred_lft forever
```

查看方式

可通过系统内查看网口数量判断，每张实际网卡系统内对应的是两个网口，系统内显示 2 个网口的，一般是 eth0, eth1 组 bond0，就无法自定义 vlan 网络，如果系统内显示 4 个网口，则可进行自定义 vlan 网络配置。

4.5 密钥与密码

本章节主要介绍密钥与密码。

创建密钥对

操作场景

为安全起见，物理机登录时建议使用密钥方式进行身份验证。因此，您需要使用已有密钥对或新建一个密钥对，用于远程登录身份验证。

新建密钥对

如果没有可用的密钥对，需要新建一个密钥对，生成公钥和私钥，并在登录物理机时进行鉴权。创建密钥对的方法如下：

- 通过管理控制台创建密钥对，公钥自动保存在系统中，私钥由用户保存在本地，具体操作请参见下文通过管理控制台创建密钥对。
- 通过 `puttygen.exe` 工具创建密钥对，公钥和私钥均保存在用户本地，具体操作请参见通过 `puttygen.exe` 工具创建密钥对。

使用已有密钥对

如果本地已有密钥对，可以直接选择管理控制台的“导入密钥对 > 选择文件”导入密钥，具体操作请参见本节导入密钥对部分。

约束与限制

- 当前支持的加解密算法为：
 - SSH-2 (RSA, 1024)
 - SSH-2 (RSA, 2048)
 - SSH-2 (RSA, 4096)
- 私钥是保证您的物理机安全的重要手段之一，用于 SSH 登录身份认证，为保证物理机安全，只能下载一次，请妥善保管

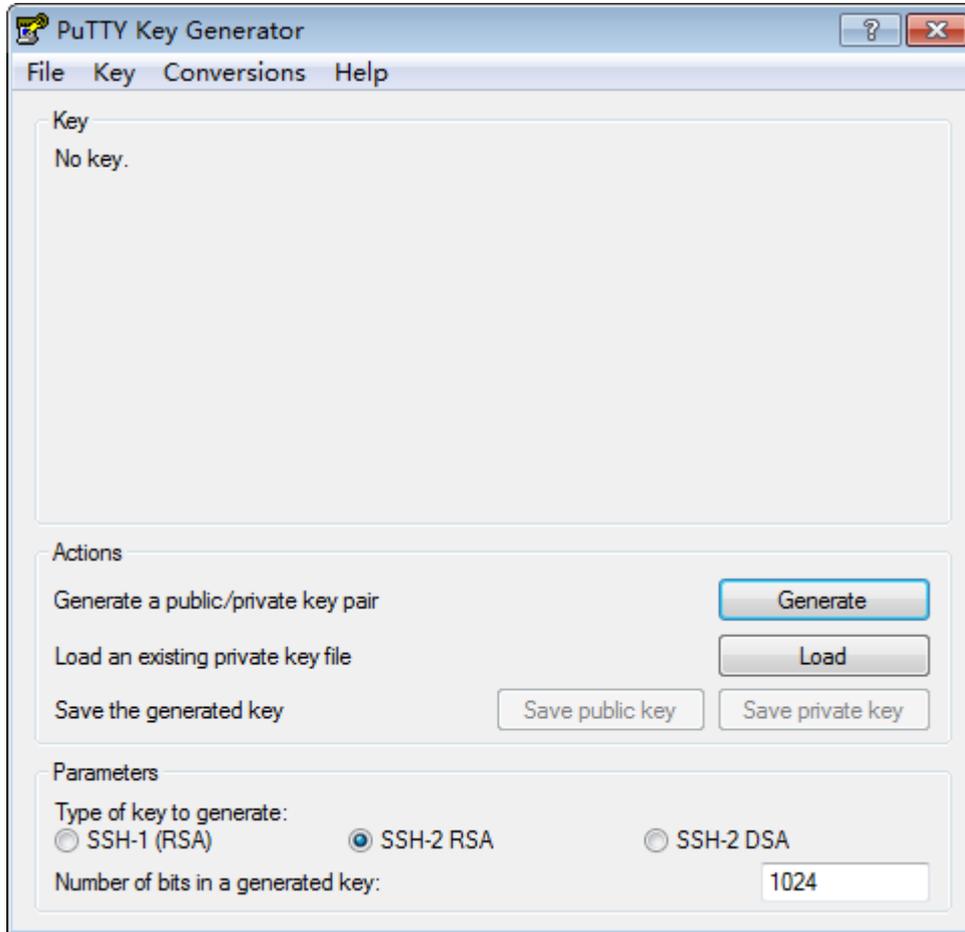
通过管理控制台创建密钥对

1. 登录管理控制台。
2. 单击管理控制台右上角的 ，选择地域。
3. 选择“计算 > 物理机”。
4. 在左侧导航树中，选择“密钥对”。
5. 在“密钥对”页面，单击“创建密钥对”。
6. 输入密钥名称。
7. 单击“确定”。
8. 在系统弹出的提示框中单击“确定”。

请根据提示信息，查看并保存私钥。为保证安全，私钥只能下载一次，请妥善保管。

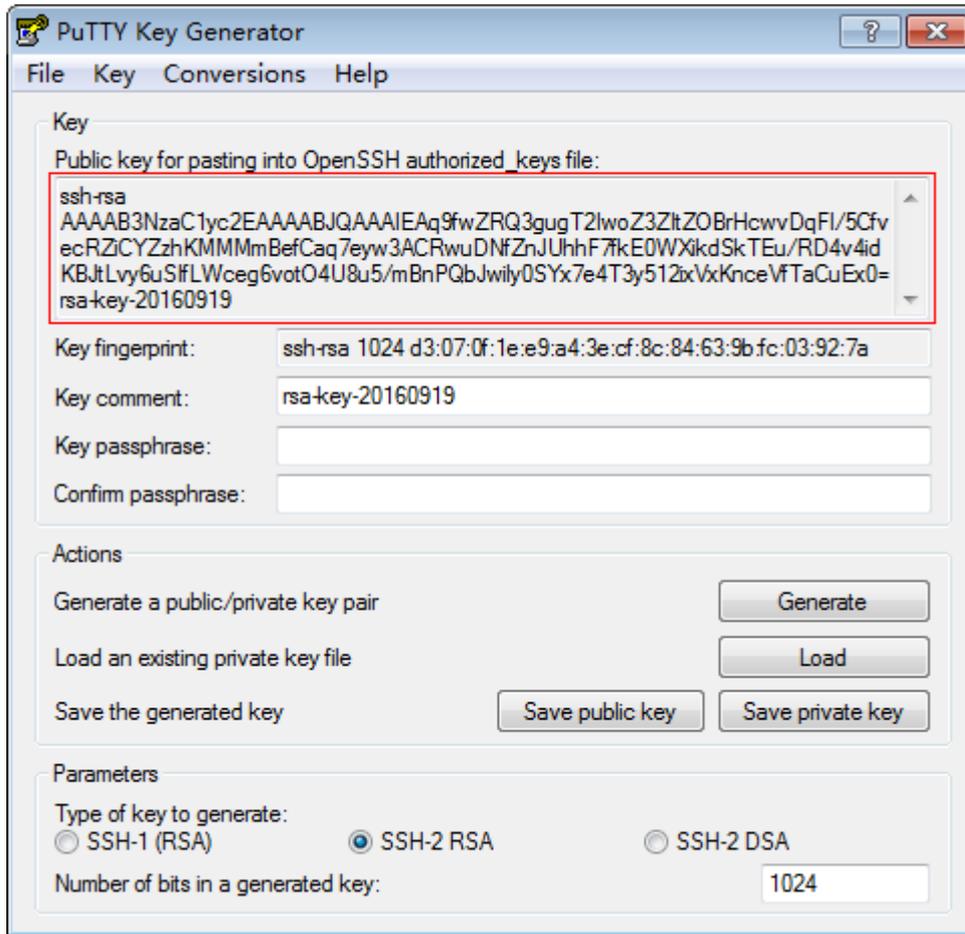
通过 `puttygen.exe` 工具创建密钥对

1. 生成公钥和私钥文件。
2. 双击“PUTTYGEN.EXE”，打开“PuTTY Key Generator”。



3. 单击“Generate”。

密钥生成器将自动生成一对公钥和私钥，其中，下图的红框内容为生成的公钥文件。



4. 复制红框中的公钥内容，并将其粘贴在文本文件中，以 .txt 格式保存在本地，保存公钥文件。

说明：

请勿直接单击“Save public key”保存公钥文件。因为通过 puttygen.exe 工具的“Save public key”按钮保存在本地的公钥文件，公钥内容的格式会发生变化，不能直接导入管理控制台使用。

5. 保存私钥文件。

根据不同用途，需将私钥文件保存为不同格式。为保证安全，私钥只能下载一次，请妥善保管。

- 保存私钥文件为“.ppk”格式。

当用户使用 PuTTY 工具登录 Linux 物理机时，需使用“.ppk”格式的私钥文件。保存方法如下：

- A. 在“PuTTY Key Generator”界面，选择“File > Save private key”。
- B. 保存私钥到本地。例如：kp-123.ppk。

- 保存私钥文件为“.pem”格式。

当用户使用 Xshell 工具登录 Linux 物理机时，或者获取 Windows 物理机的密码时，需使用 “.pem” 格式的私钥文件。保存方法如下：

- A. 在 “PuTTY Key Generator” 界面，选择 “Conversions > Export Open SSH key”。

注意：

如果该私钥文件用于 Windows 物理机的获取密码操作，在选择 “Export OpenSSH key” 时，请勿填写 “Key passphrase” 信息，否则会导致获取密码失败。

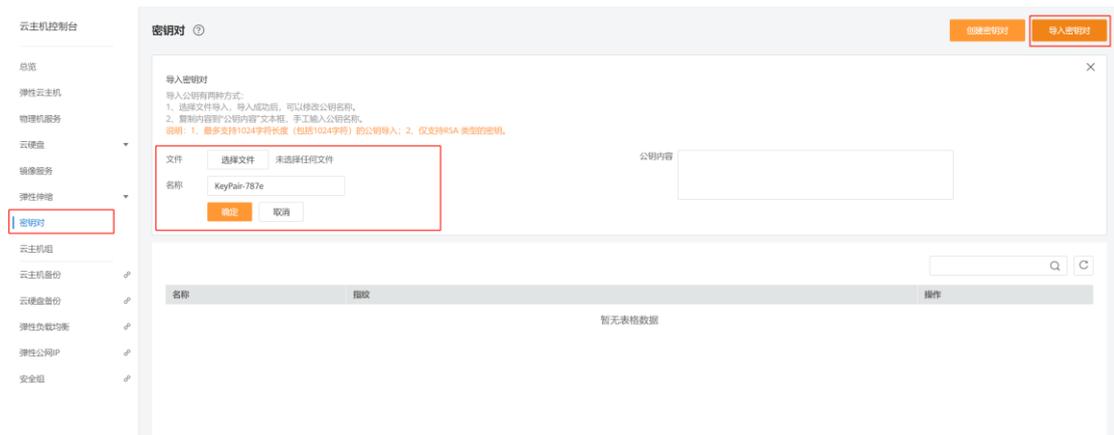
- B. 保存私钥到本地。例如：kp-123.pem。

6. 公钥和私钥文件按照需要正确保存后，请参见 “复制公钥文件内容方式”，将公钥导入系统中。

导入密钥对

通过 puttygen.exe 工具的 “Save public key” 按钮保存在本地的公钥文件，公钥内容的格式会发生变化，不能直接导入管理控制台。请先参考[通过 puttygen.exe 工具创建的密钥对，导入管理控制台失败怎么办？](#)，获取正确格式的公钥文件内容，然后再导入管理控制台。

1. 登录管理控制台。
2. 选择 “计算 > 物理机”。
3. 在左侧导航树中，选择 “密钥对”。
4. 在 “密钥对” 页面，单击 “导入密钥对”。



5. 导入密钥对的方式有如下两种：

-方式 1：选择文件方式：将本地已有的可用密钥，导入到系统中。

- a. 在管理控制台 “导入密钥对” 页面，单击 “选择文件”，选择本地保存的公钥文件。

说明：

执行“导入密钥对”操作时，请确认导入的是公钥文件，否则会导入失败。

- b. 单击“确定”，导入公钥。

公钥导入成功后，您可以修改公钥名称。

-方式 2: 复制公钥文件内容方式：将“.txt”格式文本文档中的公钥内容粘贴至对话框，导入到系统中。

- a. 复制文本文档中保存的公钥文件内容，并将其粘贴在“公钥内容”的空白区域。
- b. 单击“确定”，导入公钥。

4.6 管理弹性 IP

4.6.1 查看弹性 IP

本章节主要介绍如何查看弹性 IP。

1. 登录管理控制台。
2. 单击管理控制台左上角的，选择地域和项目。
3. 在系统首页，单击“网络 > 虚拟私有云”。
4. 单击“弹性 IP”，显示弹性 IP 列表。
5. 单击“查看弹性 IP”，跳转至弹性 IP 界面。

4.6.2 绑定弹性 IP

本章节主要介绍如何绑定弹性 IP。

当您需要为物理机器绑定弹性 IP 时，执行以下操作：

1. 登录管理控制台。
2. 单击管理控制台左上角的，选择地域和项目。
3. 选择“计算 > 物理机服务”。进入物理机器页面。
4. 单击申请的物理机器。进入详情页面。
5. 单击“弹性 IP”。根据是否有可用的弹性 IP，执行不同的操作：

若有可用的弹性 IP，执行 9~10。

若无可用的弹性 IP，执行 6~10。

6. 单击“查看弹性 IP”。跳转至弹性 IP 地址界面。
7. 单击“购买弹性 IP”，确认需要购买的 1 弹性 IP 属性。
8. 单击“提交”完成申请。返回至物理机器详情页面。
9. 单击“绑定弹性 IP”。弹出绑定弹性 IP 界面。
10. 选中要绑定的弹性 IP，单击“确定”，完成绑定。

说明：

一个网卡只能绑定一个弹性 IP。

4.6.3 解绑弹性 IP

本章节主要介绍如何解绑弹性 IP。

当您需要对绑定在物理机器上的弹性 IP 解绑定时，执行以下操作：

1. 登录管理控制台。
2. 单击管理控制台左上角的 ，选择地域和项目。
3. 选择“计算 > 物理机服务”。进入物理机器页面。
4. 单击申请的物理机器。进入详情页面。
5. 单击“弹性 IP”，找到对应的需要解绑定的弹性 IP，单击“解绑”，在弹出的提示界面单击“确定”完成解绑定操作。

4.7 配置安全组

4.7.1 添加安全组规则

本章节主要介绍如何添加安全组规则。

安全组的默认规则是在出方向上的数据报文全部放行，安全组内的物理机器无需添加规则即可互相访问。安全组创建后，您可以在安全组中定义各种访问规则，当物理机器加入该安全组后，即受到这些访问规则的保护。

说明：

在创建物理机器时只能添加一个安全组。物理机器创建完成后，可以在详情页面修改每个网卡对应的安全组。

当需要从安全组外访问安全组内的物理机器时，需要为安全组添加相应的入方向规则。建议将不同公网访问策略的物理机器划分到不同的安全组。

说明：

源地址默认的 IP 地址 0.0.0.0/0 是指允许所有 IP 地址访问安全组内的物理机器。

1. 登录管理控制台。
2. 单击管理控制台左上角的 ，选择地域和项目。
3. 在系统首页，单击“计算 > 物理机服务”。
4. 在物理机器列表，单击待变更安全组规则的物理机器名称，系统跳转至该物理机器详情页面。
5. 选择“安全组”页签，并单击 ，查看安全组规则。
6. 单击安全组 ID，系统自动跳转至安全组页面。
7. 单击“添加规则”，添加安全组规则，相关参数说明如下表所示。

参数	说明	取值样例
协议	网络协议，取值范围为：TCP，UDP，ICMP，Any。	TCP
方向	安全组规则生效的方向，取值范围：入方向，出方向。 入方向指从外表进入安全组的流量，出方向指从安全组内出去的流量。	Inbound
端口范围	规则的端口范围，取值范围为：0~65535。	22 或 22-30
源地址	当方向为入方向时，需要填入此参数。源地址可以是 IP 地址，也可以是安全组。	0.0.0.0/0 default
目的地址	当方向是出方向时，需要填入此参数。目的地址可以是 IP 地址，也可以是安全组。	0.0.0.0/0 default

4.7.2 物理机器的安全组配置案例

本章节主要介绍了不同安全组内物理机内网互通、仅允许特定 IP 地址远程登录物理机、允许任意公网 IP 地址远程登录物理机的安全组配置案例。

实际应用中，请根据实际网络环境需求，对安全组进行配置。我们提供以下几种常见的安全组配置案例作为参考。

案例一：不同安全组内的物理机器内网互通

场景举例：

在相同 Region 的相同帐号下，用户需要将某个安全组内一台物理机器上的资源拷贝到另一个安全组内的物理机器上时，可以将两台物理机器设置为内网互通后再拷贝资源。

安全组配置方法：

由于相同 Region 的相同帐号下，在同一个安全组内的物理机器默认互通，无需配置。但是，在不同安全组内的物理机器默认无法通信，此时需要添加安全组规则，使得不同安全组内的物理机器内网互通。

在两台物理机器所在安全组中分别添加一条入方向安全组规则，放通来自另一个安全组内的实例的访问，实现内网互通，安全组规则如下所示。

协议	方向	端口范围/ICMP 协议类型	源地址
设置内网互通时使用的协议类型（支持 TCP/UDP/ICMP/ANY）	入方向	设置端口范围或者 ICMP 协议类型	IPv4 地址、或者另一个安全组的 ID

案例二：仅允许特定 IP 地址远程登录物理机

场景举例：

为了防止物理机器被网络攻击，用户可以修改远程登录端口号，并设置安全组规则只允许特定的 IP 地址远程登录到物理机器。

安全组配置方法：

以仅允许特定 IP 地址（例如，192.168.20.2）通过 TCP 协议访问 Linux 操作系统的物理机器的 22 端口为例，安全组规则如下所示。

协议	方向	端口范围	源地址
TCP	入方向	22	IPv4 地址、IPv4 CIDR 或者另一个安全组的 ID。

协议	方向	端口范围	源地址
			例如：192.168.20.2

案例三：允许任意公网 IP 地址远程登录物理机器

场景举例：

允许 Internet 的任意地址远程登录到安全组内的物理机器。

安全组配置方法：

以允许任意公网地址通过 TCP 协议登录 Linux 操作系统的物理机器为例，安全组规则如下所示。

协议	方向	端口范围	源地址
TCP	入方向	22	0.0.0.0/0

以允许任意公网地址通过 TCP 协议登录 Windows 操作系统的物理机器为例，安全组规则如下所示。

协议	方向	端口范围	源地址
TCP	入方向	3389	0.0.0.0/0

案例四：允许公网地址通过 HTTP 或者 HTTPS 协议访问物理机器

场景举例：

在物理机器上部署网站后，允许 Internet 的任意 IP 地址的用户通过 HTTP 或者 HTTPS 协议访问安全组内的物理机器。

安全组配置方法：

不限制公网地址访问物理机器，需要配置的两条安全组规则如下所示。

协议	方向	端口范围	源地址
TCP	入方向	80 (HTTP)	0.0.0.0/0
TCP	入方向	443 (HTTPS)	0.0.0.0/0

4.8 管理虚拟 IP 地址

本章节主要介绍如何管理虚拟 IP 地址。

您可以绑定虚拟 IP 地址，为网卡提供额外的 IP 地址，从而实现更灵活的网络功能。

1. 登录管理控制台。

2. 单击管理控制台左上角的 ，选择地域。

3. 选择“计算 > 物理机”。

进入物理机页面。

4. 单击待绑定虚拟 IP 地址的物理机的名称。

系统跳转至该物理机的详情页面。

5. 选择“网卡”页签，并单击“管理虚拟 IP 地址”。

系统跳转至虚拟私有云页面。

6. 在“虚拟 IP”页签，您可以在列表中选择合适的虚拟 IP，或者单击“申请虚拟 IP 地址”创建新的虚拟 IP 地址。

7. 单击“操作”列下的“绑定服务器”，选择目标物理机与对应的网卡，绑定虚拟 IP 地址。

4.9 设置网卡的源/目的检查

本章节主要介绍如何设置网卡的源/目的检查。

1. 登录管理控制台。

2. 单击管理控制台左上角的 ，选择地域。

3. 选择“计算 > 物理机”。

进入物理机页面。

4. 单击待设置网卡的物理机的名称。

系统跳转至该物理机的详情页面。

5. 选择“网卡”页签，单击待设置网卡前的图标。

6. 选择打开或关闭“源/目的检查”开关。

默认情况下，“源/目的检查”状态为“ON”，系统会检查物理机发送的报文中源 IP 地址是否正确，否则不允许物理机发送该报文。这有助于防止伪装报文攻击，提升安全性。当物理机作为 NAT 服务器、路由器或防火墙时，此开关应该设置为“OFF”。

4.10 管理高速网络

4.10.1 使用限制

本章节主要介绍了高速网络的使用限制。

使用限制

- 申请物理机时，普通网卡子网的网段与高速网络的网段不能存在交集。
- 高速网络不支持安全组、EIP、DNS、VPN、专线等功能。
- 同一物理机的多个高速网卡，其所在的高速网络不能重复。
- 物理机下发成功后不能再配置高速网络。

4.10.2 创建高速网络

本章节主要介绍如何创建高速网络。

创建高速网络

1. 登录管理控制台。
2. 单击管理控制台左上角的 ，选择地域和项目。
3. 选择“计算 > 物理机”。

进入物理机页面。

4. 选择“高速网络 > 创建高速网络”。
5. 输入高速网络的名称、子网信息，单击“确定”完成申请。

4.10.3 修改高速网络名称

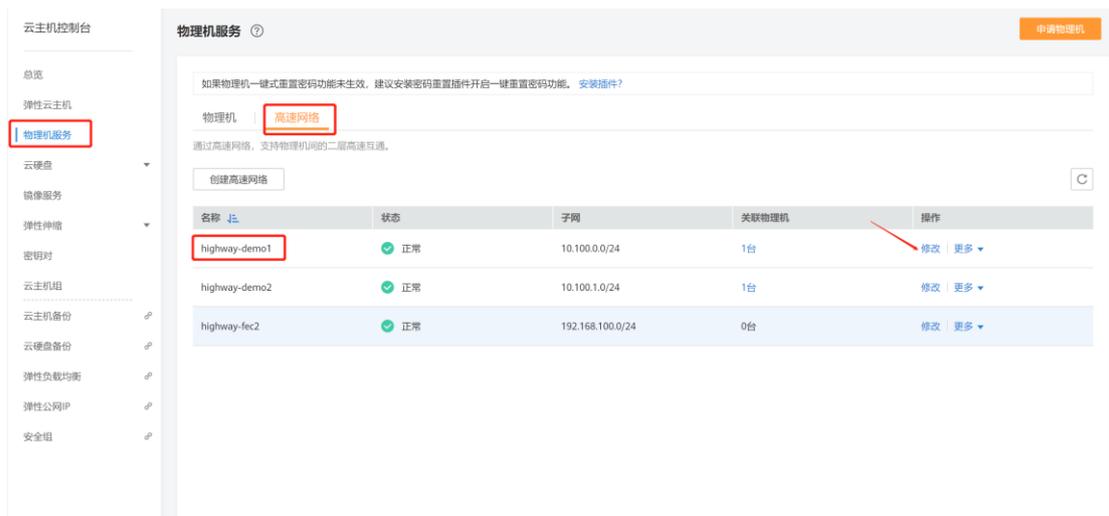
本章节主要介绍如何修改高速网络名称。

修改高速网络名称

1. 登录管理控制台。
2. 单击管理控制台的区域按钮，选择地域和项目。
3. 选择“计算 > 物理机”。

进入物理机页面。

4. 单击“高速网络”，选择准备修改的高速网络，单击“编辑”。
5. 修改高速网络的名称，单击“确定”完成修改。



4.10.4 管理私有 IP 地址

本章节主要介绍如何管理私有 IP 地址。

管理私有 IP 地址

1. 登录管理控制台。
2. 单击管理控制台右上角的区域按钮，选择地域和项目。
3. 选择“计算 > 物理机”。

进入物理机页面。

4. 单击“高速网络”，选择要配置的高速网络，单击“管理私有 IP 地址”。
 - 若需要在高速网络内预留一个私有 IP，用于申请物理机时指定该 IP 绑定高速网卡或其他用途，执行 5~6。
 - 若需要删除私有 IP 地址，执行 7。
5. 单击“分配私有 IP 地址”。
 - 若选择“自动分配”，系统将自动分配一个私有 IP 地址。
 - 若选择“手动分配”，您可以在该高速网络的 IP 范围内自己指定某个 IP 作为私有 IP 地址。
6. 单击“确定”完成操作。
7. 在要删除的私有 IP 的操作栏，单击“删除”，弹出提示界面，单击“确定”完成私有 IP 地址的删除。

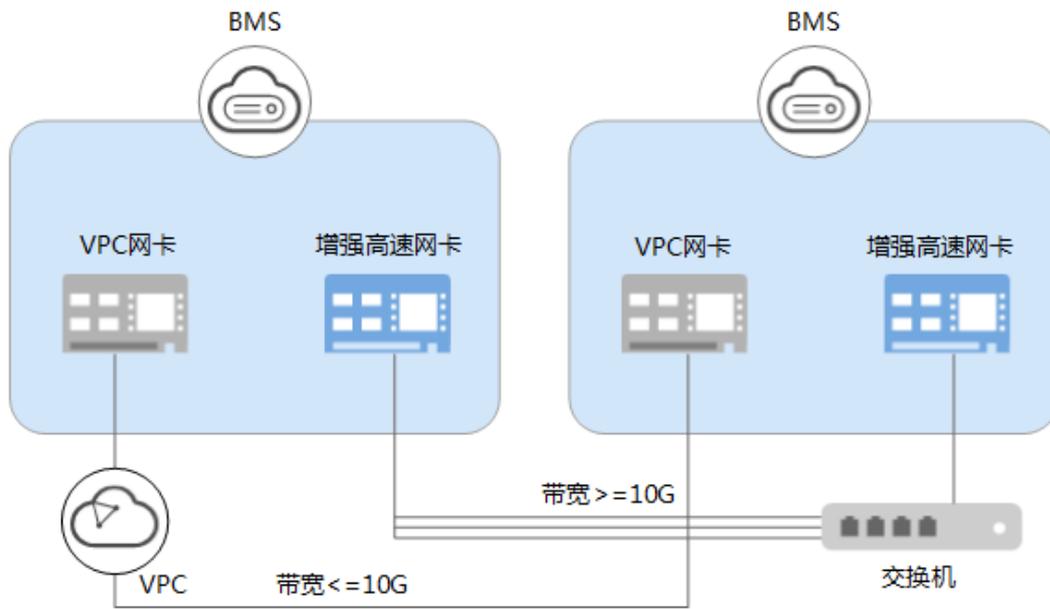
4.11 增强高速网络

4.11.1 增强高速网络概述

本章主要介绍增强高速网络。

增强高速网络

增强高速网络架构图：



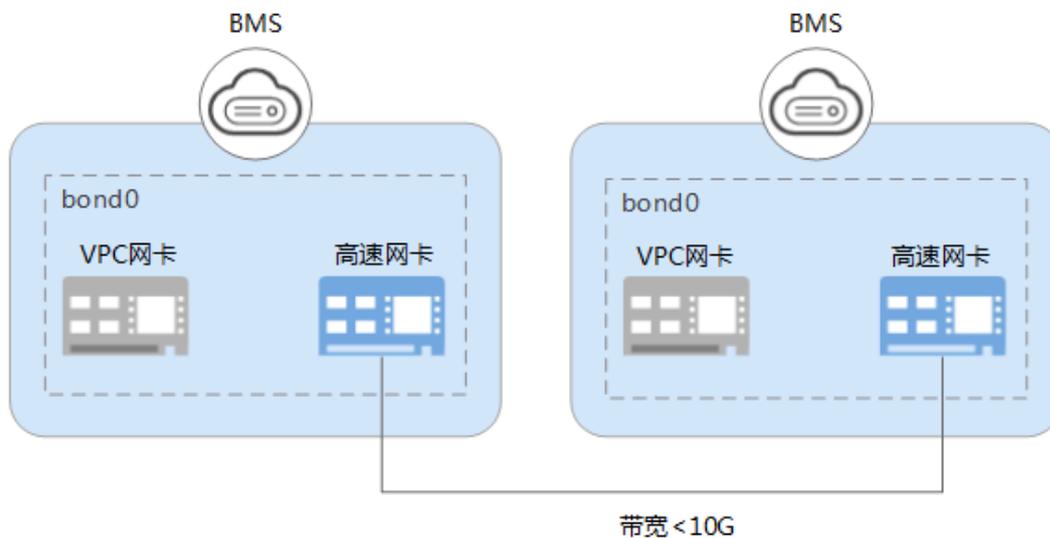
增强高速网络通过云数据中心实现内网互通互连，可以提供高质量、高速度、低时延的内网环境。具有如下特点：

- 高速带宽内部互连网络
- 客户可灵活自定义的内部网络
- 端口总带宽大于 10GE

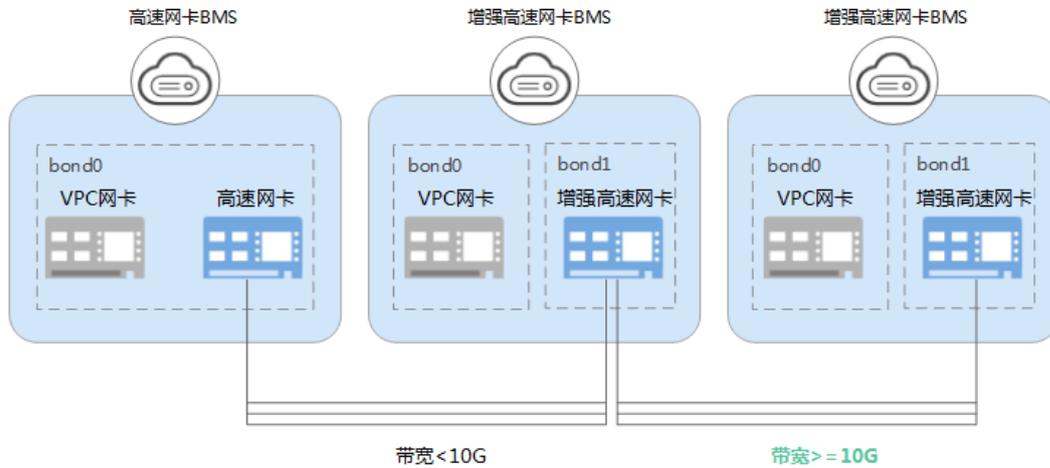
增强高速网络基于上一代高速网络进行了软硬件的优化升级。下图分别介绍上一代高速网络架构，以及高速网络和增强高速网络对比。

上一代高速网络架构图：

高速网络



两代高速网络对比图：



相比上一代高速网络，增强高速网络具有如下三大优势：

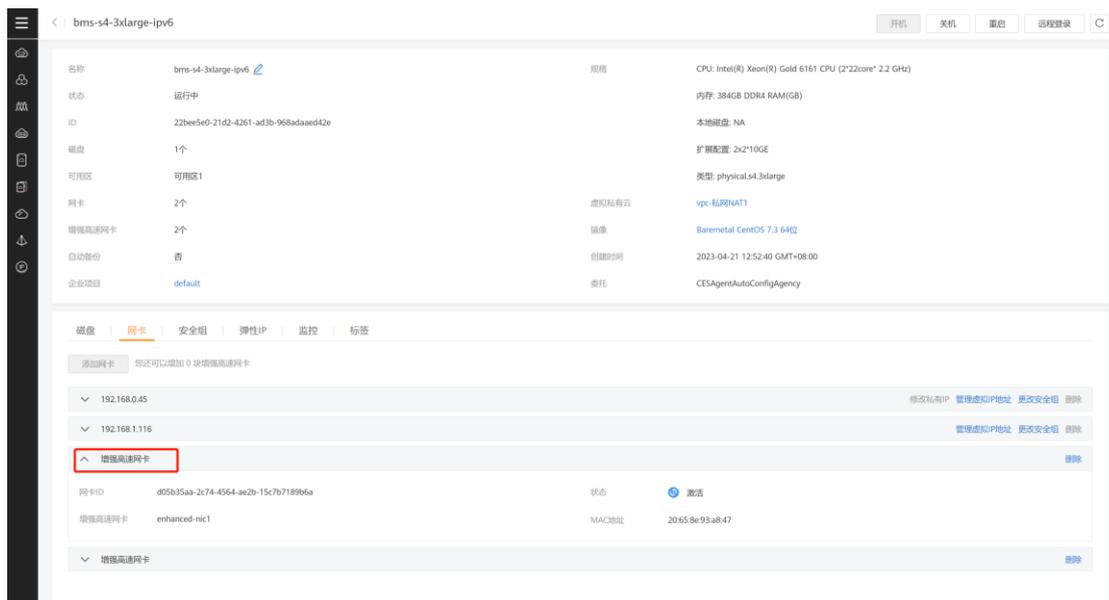
- 带宽提升至 10GE 及以上。
- 租户自定义网络平面数量，最多支持 4K 个子网。

📖 说明

目前支持的区域有：长沙 2、广州 4 和成都 3。

增强高速网络查看方式

增强高速网络的网络接口（即增强高速网卡）可以在管理控制台查看（物理机详情页“网卡”页签下）。



截图显示了云管理控制台中的实例详情页面。实例名称为 bms-s4-3xlarge-ipv6。在“网卡”页签下，可以看到已配置的网络接口。其中，一个名为“增强高速网卡”的接口被红色框选中。该接口的网卡ID为 d05k35aa-2c74-4564-ae2b-15c7b7189b6a，状态为“激活”，MAC地址为 20:65:9e:93:a8:47。

增强高速网卡使用场景

增强高速网卡主要有以下使用场景：

- 场景一：组 bond

组 bond 使用时，可以根据网络规划需求选择不配置 vlan 或者配置 vlan。

- 不配置 vlan

如果不需要 vlan，则可以在配置 bond 口时，直接配置 IP 地址和子网掩码。配置完成后，同一网络下的增强高速网卡即可相互通信。

- 配置 vlan

如果网络规划需要配置 vlan，则可以在配置 bond 口后，再配置相应 vlan 子接口。

- 场景二：不组 bond

不组 bond 直接使用时，不可以配置 vlan，直接配置 IP 地址和子网掩码。配置完成后，同一网络下的增强高速网卡即可相互通信。

📖 说明

一张增强高速网卡也可以组 bond 使用。

配置增强高速网卡(SUSE Linux Enterprise Server 12 系列)~配置增强高速网卡(Windows Server 系列) 章节举例说明增强高速网卡在 OS 内部组 bond 的配置方法，不同 OS 对应的配置方法不相同，请按相应 OS 的指导方法操作。

4.11.2 添加增强高速网卡

为物理机添加一块增强高速网卡，请参考下面步骤操作。

约束与限制：

物理机必须为“运行中”状态。

操作步骤：

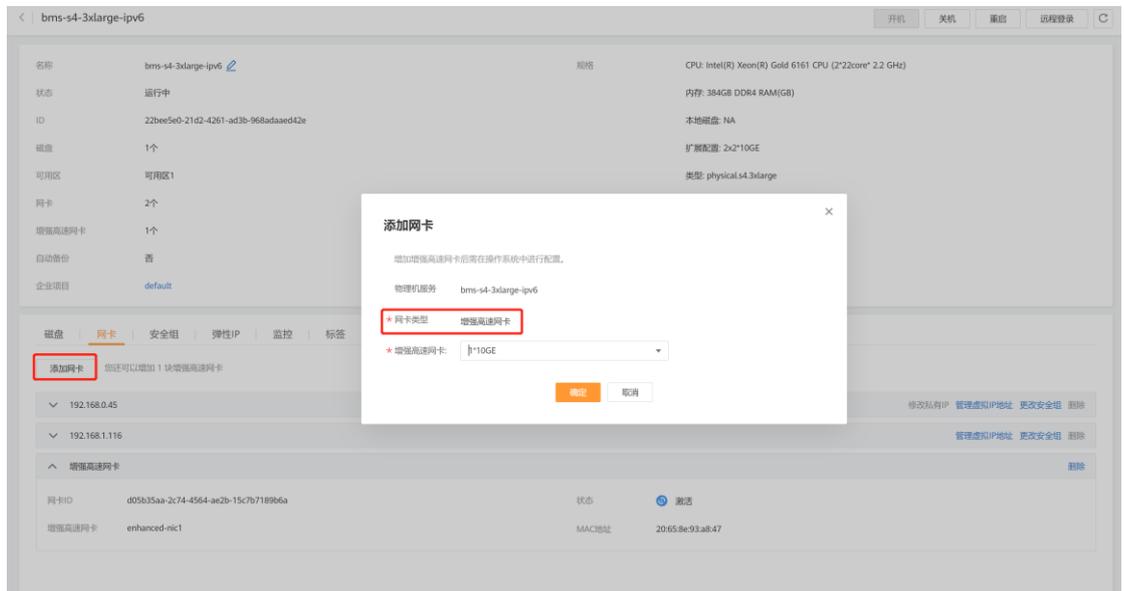
📖 说明

一台物理机最多有两块增强高速网卡，并且依赖于扩展网卡总带宽。例如，扩展网卡总带宽为 2*10GE，如果第一块增强高速网卡的带宽为 2*10GE，那么您不能再添加第二块增强高速网卡。

扩展网卡总带宽可以在规格的“扩展配置”项查看。

- 若扩展配置中包含“2*10GE”（例如 physical.h2.large，扩展配置为“1*100G IB + 2*10GE”），表示该规格物理机为单网卡，无扩展网卡，扩展网卡的带宽总量为 0。

- 若扩展配置中包含“2 x 2*10GE”（例如 physical.s3.large，扩展配置为“2 x 2*10GE”），表示该规格物理机为双网卡，有一张扩展网卡，扩展网卡的带宽总量为 2*10GE。
1. 登录管理控制台。
 2. 选择“计算 > 物理机服务”。
- 进入物理机页面。
3. 单击待添加网卡的物理机的名称。
- 系统跳转至该物理机的详情页面。
4. 选择“网卡”页签，并单击“添加网卡”。
 5. 在弹出的对话框中选择网卡类型为“增强高速网卡”，然后选择带宽大小。
 6. 单击“确定”。



后续任务：

物理机无法识别新添加的增强高速网卡，需要参考配置增强高速网卡（SUSE Linux Enterprise Server 12 系列）~ 配置增强高速网卡（Windows Server 系列）手动激活网卡。

4.11.3 删除增强高速网卡

操作场景：

如果不再需要增强高速网卡，请参考此章节进行删除。

约束与限制：

物理机必须为“运行中”或“关机”状态。

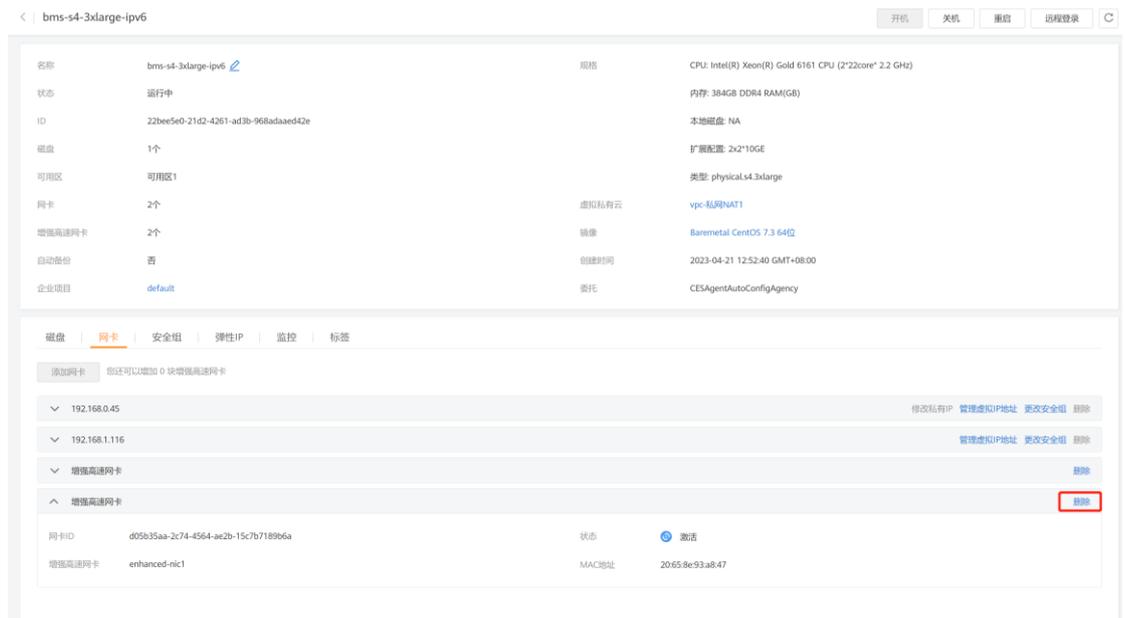
操作步骤:

1. 登录管理控制台。
2. 选择“计算 > 物理机服务”。
进入物理机页面。
3. 单击待删除网卡的物理机的名称。
系统跳转至该物理机的详情页面。
4. 选择“网卡”页签，单击待删除增强高速网卡前的下拉图标，记录 VLAN 和 MAC 地址。

说明

界面上删除网卡后，还需要进入 BMS 操作系统内部执行相关操作，删除设备（需要使用此处记录的 VLAN 和 MAC 地址）。

5. 单击“删除”。
6. 在弹出的对话框中单击“确定”，删除网卡。



后续任务:

参考配置增强高速网卡（SUSE Linux Enterprise Server 12 系列）~ 配置增强高速网卡（Windows Server 系列）中的“删除网卡”部分删除网络设备。

4.11.4 配置增强高速网卡（SUSE Linux Enterprise Server 12 系列）

下面以 SUSE Linux Enterprise Server 12 SP3 (x86_64)操作系统为例，举例介绍物理机增强高速网卡的配置方法，包括增加和删除网卡时的配置。

增加网卡:

📖 说明

其他 OS 的配置方法请参考：

- [SUSE Linux Enterprise Server 11 系列](#)
- [RedHat 系列, CentOS 系列, Oracle Linux 系列, Euler 系列](#)
- [Ubuntu 系列](#)
- [Windows Server 系列](#)

1. 以“root”用户，使用密钥或密码登录物理机。
2. 进入物理机的命令行界面，查询网卡信息。

ip link

返回信息示例如下：

```
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN mode
DEFAULT group default qlen 1
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
2: eth0: <BROADCAST,MULTICAST,SLAVE,UP,LOWER_UP> mtu 8888 qdisc mq master
bond0 state UP mode DEFAULT group default qlen 1000
    link/ether fa:16:00:57:90:c9 brd ff:ff:ff:ff:ff:ff
3: eth1: <BROADCAST,MULTICAST,SLAVE,UP,LOWER_UP> mtu 8888 qdisc mq master
bond0 state UP mode DEFAULT group default qlen 1000
    link/ether fa:16:00:57:90:c9 brd ff:ff:ff:ff:ff:ff
4: eth2: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN mode DEFAULT
group default qlen 1000
    link/ether 40:7d:0f:52:e3:a5 brd ff:ff:ff:ff:ff:ff
5: eth3: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN mode DEFAULT
group default qlen 1000
    link/ether 40:7d:0f:52:e3:a6 brd ff:ff:ff:ff:ff:ff
6: bond0: <BROADCAST,MULTICAST,MASTER,UP,LOWER_UP> mtu 8888 qdisc noqueue
state UP mode DEFAULT group default qlen 1000
    link/ether fa:16:00:57:90:c9 brd ff:ff:ff:ff:ff:ff
```

📖 说明

其中，“eth0”和“eth1”为承载 VPC 网络的网络设备，“eth2”和“eth3”为承载增强高速网络的网络设备。

3. 设置 udev 规则。

执行以下命令创建“80-persistent-net.rules”文件。

cp /etc/udev/rules.d/70-persistent-net.rules /etc/udev/rules.d/80-persistent-net.rules

将 2 中查询到的，且“80-persistent-net.rules”中未体现的网卡 MAC 地址和名称，写入该文件中，使得物理机重启复位后，网卡名称和顺序不会发生改变。

📖 说明

网卡的 MAC 地址和名称中的字母，请使用小写字母。

vim /etc/udev/rules.d/80-persistent-net.rules

修改后的示例如下：

```
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="?*",
ATTR{address}=="f4:4c:7f:5d:b7:2a", NAME="eth0"
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="?*",
ATTR{address}=="f4:4c:7f:5d:b7:2b", NAME="eth1"
```

```
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="?*",
ATTR(address)=="40:7d:0f:52:e3:a5", NAME="eth2"
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="?*",
ATTR(address)=="40:7d:0f:52:e3:a6", NAME="eth3"
```

4. 创建“eth2”和“eth3”网卡的配置文件，可以通过复制已有网卡配置文件的方式快速创建。

```
cd /etc/sysconfig/network
```

```
cp ifcfg-eth0 ifcfg-eth2
```

```
cp ifcfg-eth1 ifcfg-eth3
```

修改“eth2”和“eth3”网卡的配置文件。

```
vi ifcfg-eth2
```

“eth2”网卡配置文件的修改示例如下：

```
STARTMODE=auto
MTU=8888
NM_CONTROLLED=no
BOOTPROTO=STATIC
DEVICE=eth2
USERCONTRL=no
LLADDR=40:7d:0f:52:e3:a5
TYPE=Ethernet
```

📖 说明

其中，参数“MTU”配置为“8888”，“BOOTPROTO”需要配置为“STATIC”，参数“DEVICE”、“LLADDR”根据实际情况填写。

```
vi ifcfg-eth3
```

“eth3”网卡配置文件的修改示例如下：

```
STARTMODE=auto
MTU=8888
NM_CONTROLLED=no
BOOTPROTO=STATIC
DEVICE=eth3
USERCONTRL=no
LLADDR=40:7d:0f:52:e3:a6
TYPE=Ethernet
```

修改完成后，保存并退出。

5. 将“eth2”和“eth3”组 bond，假设为“bond1”。

创建 ifcfg-bond1 文件并修改配置：

```
cp ifcfg-bond0 ifcfg-bond1
```

```
vi ifcfg-bond1
```

“bond1”网卡配置文件的修改示例如下：

```
BONDING MASTER=yes
TYPE=Bond
MTU=8888
STARTMODE=auto
BONDING MODULE_OPTS="mode=1 miimon=100"
NM_CONTROLLED=no
BOOTPROTO=STATIC
DEVICE=bond1
USERCONTRL=no
```

```
LLADDR=40:7d:0f:52:e3:a5
BONDING_SLAVE1=eth2
BONDING_SLAVE0=eth3
IPADDR=10.10.10.104
NETMASK=255.255.255.0
NETWORK=10.10.10.0
```

📖 说明

其中，参数“MTU”配置为“8888”，“BONDING_MODULE_OPTS”配置为“mode=1 miimon=100”，“BOOTPROTO”需要配置为“STATIC”，“DEVICE”、“BONDING_SLAVE1”、“BONDING_SLAVE0”、“IPADDR”、“NETMASK”、“NETWORK”根据实际需要填写，“LLADDR”配置为参数“BONDING_SLAVE1”对应网卡的 LLADDR。

修改完成后，保存并退出。

6. 执行以下命令，启动新增的 bond1 网卡。

wicked ifup bond1

7. 查询 IP 地址信息，可查看到 IP 地址已分配。

ip addr show

示例如下：

```
1: lo: <LOOPBACK,UP,LOWER UP> mtu 65536 qdisc noqueue state UNKNOWN group
default qlen 1
    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,SLAVE,UP,LOWER UP> mtu 8888 qdisc mq master
bond0 state UP group default qlen 1000
    link/ether fa:16:00:57:90:c9 brd ff:ff:ff:ff:ff:ff
3: eth1: <BROADCAST,MULTICAST,SLAVE,UP,LOWER UP> mtu 8888 qdisc mq master
bond0 state UP group default qlen 1000
    link/ether fa:16:00:57:90:c9 brd ff:ff:ff:ff:ff:ff
4: eth2: <BROADCAST,MULTICAST,SLAVE,UP,LOWER UP> mtu 1500 qdisc mq master
bond1 state UP group default qlen 1000
    link/ether 40:7d:0f:52:e3:a5 brd ff:ff:ff:ff:ff:ff
5: eth3: <BROADCAST,MULTICAST,SLAVE,UP,LOWER UP> mtu 1500 qdisc mq master
bond1 state UP group default qlen 1000
    link/ether 40:7d:0f:52:e3:a5 brd ff:ff:ff:ff:ff:ff
6: bond0: <BROADCAST,MULTICAST,MASTER,UP,LOWER UP> mtu 8888 qdisc noqueue
state UP group default qlen 1000
    link/ether fa:16:00:57:90:c9 brd ff:ff:ff:ff:ff:ff
    inet 172.16.2.44/24 brd 172.16.2.255 scope global bond0
        valid_lft forever preferred_lft forever
    inet6 fe80::f816:ff:fe57:90c9/64 scope link
        valid_lft forever preferred_lft forever
7: bond1: <BROADCAST,MULTICAST,MASTER,UP,LOWER UP> mtu 1500 qdisc noqueue
state UP group default qlen 1000
    link/ether 40:7d:0f:52:e3:a5 brd ff:ff:ff:ff:ff:ff
    inet 10.10.10.104/24 brd 10.10.10.255 scope global bond1
        valid_lft forever preferred_lft forever
    inet6 fe80::427d:fff:fe52:e3a5/64 scope link
        valid_lft forever preferred_lft forever
```

8. 参考上述步骤，完成其他物理机上的配置。

删除网卡：

1. 获取待删除增强高速网卡的 bond 网卡地址。
2. 以“root”用户，使用密钥或密码登录物理机。
3. 找到 bond 网络设备，然后执行以下命令关闭并删除网络设备。
wicked ifdown bond1
4. 执行以下命令，删除网络配置文件“/etc/sysconfig/network-scripts/ifcfg-eth2”、“/etc/sysconfig/network-scripts/ifcfg-eth3”和“/etc/sysconfig/network-scripts/ifcfg-bond1”。
rm -f /etc/sysconfig/network-scripts/ifcfg-eth2
rm -f /etc/sysconfig/network-scripts/ifcfg-eth3
rm /etc/sysconfig/network/ifcfg-bond1

4.11.5 配置增强高速网卡（SUSE Linux Enterprise Server 11 系列）

下面以 SUSE Linux Enterprise Server 11 SP4 操作系统为例，举例介绍物理机增强高速网卡的配置方法。

增加网卡：

1. 以“root”用户，使用密钥或密码登录物理机。
2. 进入物理机的命令行界面，查询网卡信息。

ip link

返回信息示例如下：

```
1: lo: <LOOPBACK,UP,LOWER UP> mtu 65536 qdisc noqueue state UNKNOWN mode
DEFAULT group default qlen 1
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
2: eth0: <BROADCAST,MULTICAST,SLAVE,UP,LOWER UP> mtu 8888 qdisc mq master
bond0 state UP mode DEFAULT group default qlen 1000
    link/ether fa:16:00:57:90:c9 brd ff:ff:ff:ff:ff:ff
3: eth1: <BROADCAST,MULTICAST,SLAVE,UP,LOWER UP> mtu 8888 qdisc mq master
bond0 state UP mode DEFAULT group default qlen 1000
    link/ether fa:16:00:57:90:c9 brd ff:ff:ff:ff:ff:ff
4: eth2: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN mode DEFAULT
group default qlen 1000
    link/ether 40:7d:0f:52:e3:a5 brd ff:ff:ff:ff:ff:ff
5: eth3: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN mode DEFAULT
group default qlen 1000
    link/ether 40:7d:0f:52:e3:a6 brd ff:ff:ff:ff:ff:ff
6: bond0: <BROADCAST,MULTICAST,MASTER,UP,LOWER_UP> mtu 8888 qdisc noqueue
state UP mode DEFAULT group default qlen 1000
    link/ether fa:16:00:57:90:c9 brd ff:ff:ff:ff:ff:ff
```

📖 说明

其中，“eth0”和“eth1”为承载 VPC 网络的网络设备，“eth2”和“eth3”为承载自定义 VLAN 网络的网络设备。

3. 设置 udev 规则。

执行以下命令创建“80-persistent-net.rules”文件。

```
cp /etc/udev/rules.d/70-persistent-net.rules /etc/udev/rules.d/80-persistent-net.rules
```

将 2 中查询到的，且“80-persistent-net.rules”中未体现的网卡 MAC 地址和名称，写入该文件中，使得物理机重启复位后，网卡名称和顺序不会发生改变。

📖 说明

网卡的 MAC 地址和名称中的字母，请使用小写字母。

```
vim /etc/udev/rules.d/80-persistent-net.rules
```

修改后的示例如下：

```
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="?*",
ATTR{address}=="f4:4c:7f:5d:b7:2a", NAME="eth0"
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="?*",
ATTR{address}=="f4:4c:7f:5d:b7:2b", NAME="eth1"
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="?*",
ATTR{address}=="40:7d:0f:52:e3:a5", NAME="eth2"
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="?*",
ATTR{address}=="40:7d:0f:52:e3:a6", NAME="eth3"
```

4. 创建“eth2”和“eth3”网卡的配置文件。

可通过复制已有网卡配置文件的方式快速创建。

```
cd /etc/sysconfig/network
```

```
cp ifcfg-eth0 ifcfg-eth2
```

```
cp ifcfg-eth1 ifcfg-eth3
```

修改“eth2”和“eth3”网卡的配置文件。

```
vi ifcfg-eth2
```

“eth2”网卡配置文件的修改示例如下：

```
STARTMODE=auto
MTU=8888
NM_CONTROLLED=no
BOOTPROTO=STATIC
DEVICE=eth2
USERCONTRL=no
LLADDR=40:7d:0f:52:e3:a5
TYPE=Ethernet
```

📖 说明

其中，参数参数“MTU”配置为“8888”，“BOOTPROTO”需要配置为“STATIC”，参数“DEVICE”、“LLADDR”根据实际需要填写。

```
vi ifcfg-eth3
```

“eth3”网卡配置文件的修改示例如下：

```
STARTMODE=auto
MTU=8888
NM_CONTROLLED=no
BOOTPROTO=STATIC
```

```
DEVICE=eth3
USERCONTRL=no
LLADDR=40:7d:0f:52:e3:a6
TYPE=Ethernet
```

修改完成后，保存并退出。

5. 将“eth2”和“eth3”组 bond，假设为“bond1”。

创建 ifcfg-bond1 文件并修改配置：

```
cp ifcfg-bond0 ifcfg-bond1
```

```
vi ifcfg-bond1
```

“bond1”网卡配置文件的修改示例如下：

```
BONDING MASTER=yes
TYPE=Bond
MTU=8888
STARTMODE=auto
BONDING MODULE_OPTS="mode=1 miimon=100"
NM CONTROLLED=no
BOOTPROTO=STATIC
DEVICE=bond1
USERCONTRL=no
LLADDR=40:7d:0f:52:e3:a5
BONDING_SLAVE1=eth2
BONDING_SLAVE0=eth3
IPADDR=10.10.10.104
NETMASK=255.255.255.0
NETWORK=10.10.10.0
```

说明

其中，参数“MTU”配置为“8888”，“BONDING_MODULE_OPTS”配置为“mode=1 miimon=100”，“BOOTPROTO”需要配置为“STATIC”，“DEVICE”、“BONDING_SLAVE1”、“BONDING_SLAVE0”、“IPADDR”、“NETMASK”、“NETWORK”根据实际需要填写，“LLADDR”配置为参数“BONDING_SLAVE1”对应网卡的 LLADDR。

修改完成后，保存并退出。

6. 执行以下命令，启动新增的 bond1 网卡。

```
ifup bond1
```

7. 查询 IP 地址信息，可查看到 IP 地址已分配。

```
ip addr show
```

示例如下：

```
1: lo: <LOOPBACK,UP,LOWER UP> mtu 65536 qdisc noqueue state UNKNOWN group
default qlen 1
    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,SLAVE,UP,LOWER_UP> mtu 8888 qdisc mq master
bond0 state UP group default qlen 1000
    link/ether fa:16:00:57:90:c9 brd ff:ff:ff:ff:ff:ff
3: eth1: <BROADCAST,MULTICAST,SLAVE,UP,LOWER_UP> mtu 8888 qdisc mq master
bond0 state UP group default qlen 1000
```

```
link/ether fa:16:00:57:90:c9 brd ff:ff:ff:ff:ff:ff
4: eth2: <BROADCAST,MULTICAST,SLAVE,UP,LOWER UP> mtu 1500 qdisc mq master
bond1 state UP group default qlen 1000
link/ether 40:7d:0f:52:e3:a5 brd ff:ff:ff:ff:ff:ff
5: eth3: <BROADCAST,MULTICAST,SLAVE,UP,LOWER UP> mtu 1500 qdisc mq master
bond1 state UP group default qlen 1000
link/ether 40:7d:0f:52:e3:a5 brd ff:ff:ff:ff:ff:ff
6: bond0: <BROADCAST,MULTICAST,MASTER,UP,LOWER UP> mtu 8888 qdisc noqueue
state UP group default qlen 1000
link/ether fa:16:00:57:90:c9 brd ff:ff:ff:ff:ff:ff
inet 172.16.2.44/24 brd 172.16.2.255 scope global bond0
valid_lft forever preferred_lft forever
inet6 fe80::f816:ff:fe57:90c9/64 scope link
valid_lft forever preferred_lft forever
7: bond1: <BROADCAST,MULTICAST,MASTER,UP,LOWER UP> mtu 1500 qdisc noqueue
state UP group default qlen 1000
link/ether 40:7d:0f:52:e3:a5 brd ff:ff:ff:ff:ff:ff
inet 10.10.10.104/24 brd 10.10.10.255 scope global bond1
valid_lft forever preferred_lft forever
inet6 fe80::427d:fff:fe52:e3a5/64 scope link
valid_lft forever preferred_lft forever
```

8. 参考上述步骤，完成其他物理机上的配置。

删除网卡：

1. 获取待删除增强高速网卡的 bond 网卡地址。
2. 以“root”用户，使用密钥或密码登录物理机。
3. 找到 bond 网络设备，然后执行命令关闭并删除网络设备。

ifdown bond1

4. 执行以下命令，删除网络配置文件“/etc/sysconfig/network-scripts/ifcfg-eth2”、“/etc/sysconfig/network-scripts/ifcfg-eth3”和“/etc/sysconfig/network-scripts/ifcfg-bond1”。

```
rm -f /etc/sysconfig/network-scripts/ifcfg-eth2
```

```
rm -f /etc/sysconfig/network-scripts/ifcfg-eth3
```

```
rm /etc/sysconfig/network/ifcfg-bond1
```

4.11.6 配置增强高速网卡 (RedHat 系列, CentOS 系列, Oracle Linux 系列, Euler 系列)

下面以 CentOS 6.9 (x86_64)操作系统为例，举例介绍物理机增强高速网卡的配置方法。

说明

RedHat 系列、Oracle Linux 系列、Euler 系列及 CentOS 系列操作系统的配置方法类似。

增加网卡：

以“root”用户，使用密钥或密码登录物理机。执行如下命令：

```
blkid | grep config-2
```

如果输出结果为空，请采用[配置方式二](#)；如果输出结果如下图所示，请采用[配置方式一](#)。

```
[root@bms-8d3e ~]# blkid | grep config-2
/dev/sda4: UUID="2019-04-01-16-57-22-00" LABEL="config-2" TYPE="iso9660"
```

配置方式一：

步骤 1 登录物理机服务器

以“root”用户，使用密钥或密码登录物理机。

步骤 2 进入物理机的命令行界面，查询网卡信息。

```
ip link
```

返回信息示例如下：

```
[root@bms-centos ~]# ip link
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
2: eth0: <BROADCAST,MULTICAST,SLAVE,UP,LOWER_UP> mtu 8888 qdisc mq master bond0 state UP qlen 1000
    link/ether fa:16:00:6d:80:29 brd ff:ff:ff:ff:ff:ff
3: eth1: <BROADCAST,MULTICAST,SLAVE,UP,LOWER_UP> mtu 8888 qdisc mq master bond0 state UP qlen 1000
    link/ether fa:16:00:6d:80:29 brd ff:ff:ff:ff:ff:ff
4: eth2: <BROADCAST,MULTICAST> mtu 8888 qdisc mq state DOWN qlen 1000
    link/ether 40:7d:0f:52:e3:a5 brd ff:ff:ff:ff:ff:ff
5: eth3: <BROADCAST,MULTICAST> mtu 8888 qdisc mq state DOWN qlen 1000
    link/ether 40:7d:0f:52:e3:a6 brd ff:ff:ff:ff:ff:ff
6: bond0: <BROADCAST,MULTICAST,PROMISC,MASTER,UP,LOWER_UP> mtu 8888 qdisc noqueue state UP
    link/ether fa:16:00:6d:80:29 brd ff:ff:ff:ff:ff:ff
[root@bms-centos ~]#
```

说明

其中，“eth0”和“eth1”为承载 VPC 网络的网络设备，“eth2”和“eth3”为承载增强高速网络的网络设备。

步骤 3 检查 udev 规则配置文件

执行以下命令，查看“/etc/udev/rules.d/”目录下是否有“80-persistent-net.rules”配置文件。

```
ll /etc/udev/rules.d/ | grep 80-persistent-net.rules
```

- 如果存在“80-persistent-net.rules”，且该配置文件中已存在[步骤 2](#)中查询到的除“bond0”和“lo”以外的其它所有网卡和对应的 MAC 地址，请执行[步骤 6](#)。
- 否则，继续执行[步骤 4](#)。

步骤 4 拷贝 udev 规则配置文件

执行以下命令，将“/etc/udev/rules.d/70-persistent-net.rules”文件拷贝一份（文件名为“/etc/udev/rules.d/80-persistent-net.rules”）。

```
cp -p /etc/udev/rules.d/70-persistent-net.rules /etc/udev/rules.d/80-persistent-net.rules
```

说明

如果没有“/etc/udev/rules.d/70-persistent-net.rules”文件，请直接创建“/etc/udev/rules.d/80-persistent-net.rules”文件，按如下格式填写内容：

```
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="?*",
ATTR{address}=="4c:f9:5d:d9:e8:ac", NAME="eth0"
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="?*",
ATTR{address}=="4c:f9:5d:d9:e8:ad", NAME="eth1"
```

步骤 5 设置 udev 规则。

将[步骤 2](#) 中查询到的除“eth0”和“eth1”以外的网卡（即“/etc/udev/rules.d/70-persistent-net.rules”中未体现的网卡 MAC 地址和名称），写入“/etc/udev/rules.d/80-persistent-net.rules”文件中，使得物理机重启复位后，网卡名称和顺序不会发生改变。

说明

网卡的 MAC 地址和名称中的字母，请使用小写字母。

vi /etc/udev/rules.d/80-persistent-net.rules

修改后的示例如下：

```
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="?*", ATTR{address}=="f4:4c:7f:5d:b7:2a", NAME="eth0"
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="?*", ATTR{address}=="f4:4c:7f:5d:b7:2b", NAME="eth1"
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="?*", ATTR{address}=="40:7d:0f:52:e3:a5", NAME="eth2"
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="?*", ATTR{address}=="40:7d:0f:52:e3:a6", NAME="eth3"
```

修改完成后，按“Esc”，输入:wq 保存并退出。

步骤 6 配置网络配置文件

执行以下命令，将网络配置文件“/etc/sysconfig/network-scripts/ifcfg-bond0”拷贝为“/etc/sysconfig/network-scripts/ifcfg-bond1”，将网络配置文件“/etc/sysconfig/network-scripts/ifcfg-eth0”拷贝为“/etc/sysconfig/network-scripts/ifcfg-eth2”和“/etc/sysconfig/network/ifcfg-eth3”。

```
cp -p /etc/sysconfig/network-scripts/ifcfg-bond0 /etc/sysconfig/network-
scripts/ifcfg-bond1
cp -p /etc/sysconfig/network-scripts/ifcfg-eth0 /etc/sysconfig/network-
scripts/ifcfg-eth2
cp -p /etc/sysconfig/network-scripts/ifcfg-eth0 /etc/sysconfig/network-
scripts/ifcfg-eth3
```

步骤 7 修改网络配置文件

执行以下命令，编辑“/etc/sysconfig/network-scripts/ifcfg-eth2”和“/etc/sysconfig/network-scripts/ifcfg-eth3”，配置“eth2”设备和“eth3”设备的网络配置文件。

- **vi /etc/sysconfig/network-scripts/ifcfg-eth2**

“eth2”按以下格式编辑：

```
USERCTL=no
MTU=8888
NM CONTROLLED=no
BOOTPROTO=static
DEVICE=eth2
```

```
TYPE=Ethernet
ONBOOT=yes
MASTER=bond1
SLAVE=yes
```

其中，“BOOTPROTO”参数取值修改为“static”，“DEVICE”为对应的网络设备名称，取值即为“eth2”，“MASTER”为对应的增强高速网卡 bond 的名称，取值如“bond1”，其他参数可保持不变。

- **vi /etc/sysconfig/network-scripts/ifcfg-eth3**

“eth3”按以下格式编辑（格式和规则和“eth2”一致）：

```
USERCTL=no
MTU=8888
NM_CONTROLLED=no
BOOTPROTO=static
DEVICE=eth3
TYPE=Ethernet
ONBOOT=yes
MASTER=bond1
SLAVE=yes
```

步骤 8 执行以下命令，编辑“/etc/sysconfig/network-scripts/ifcfg-bond1”。

- **vi /etc/sysconfig/network-scripts/ifcfg-bond1**

按以下格式编辑：

```
MACADDR=40:7d:0f:52:e3:a5
BONDING MASTER=yes
USERCTL=no
ONBOOT=yes
NM_CONTROLLED=no
BOOTPROTO=static
BONDING_OPTS="mode=1 miimon=100"
DEVICE=bond1
TYPE=Bond
IPADDR=10.10.10.101
NETMASK=255.255.255.0
MTU=8888
```

其中，

- “MACADDR”参数取值修改为增强高速网卡“eth2”或者“eth3”设备的 MAC 地址。
- “BOOTPROTO”参数取值修改为“static”。
- “DEVICE”参数取值修改为“bond1”。
- “IPADDR”参数取值修改为待给“bond1”分配的 IP 地址（为增强高速网络规划的 IP 地址在没有与 VPC 网段冲突的情况下可任意规划，需要通过增强高速网络通信的物理机须将增强高速网络配置在同一个网段），如“10.10.10.101”。
- “NETMASK”参数为给增强高速网络“bond1”配置的 IP 的子网掩码。

其他参数可保持不变。

修改完成后，按“Esc”，输入:wq 保存并退出。

步骤 9 执行以下命令，启动增强高速网络端口组“bond1”。

首先启动增强高速网卡“eth2”和“eth3”设备。

```
ifup eth2
```

```
ifup eth3
```

```
ifup bond1
```

```
[root@bms-centos network-scripts]# ifup bond1  
Determining if ip address 10.10.10.101 is already in use for device bond1...
```

步骤 10 参见上述步骤，完成其他物理机的配置。

步骤 11 待其他物理机配置完成后，互相 ping 对端增强高速网络配置的同网段 IP，检查是否可以 ping 通。

```
[root@bms-centos network-scripts]# ping 10.10.10.102 -I bond1  
PING 10.10.10.102 (10.10.10.102) from 10.10.10.101 bond1: 56(84) bytes of data:  
64 bytes from 10.10.10.102: icmp_seq=1 ttl=64 time=0.475 ms  
64 bytes from 10.10.10.102: icmp_seq=2 ttl=64 time=0.033 ms  
64 bytes from 10.10.10.102: icmp_seq=3 ttl=64 time=0.032 ms  
^C  
--- 10.10.10.102 ping statistics ---  
3 packets transmitted, 3 received, 0% packet loss, time 2263ms  
rtt min/avg/max/mdev = 0.032/0.180/0.475/0.208 ms
```

---结束

配置方式二：

步骤 1 以“root”用户，使用密钥或密码登录物理机。

步骤 2 进入物理机的命令行界面，查询网卡信息。

```
ip link
```

返回信息示例如下：

```
[root@bms-b467 ~]# ip link
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN mode DEFAULT group default qlen 1000
   link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP mode DEFAULT group default qlen 1000
   link/ether fa:16:3e:23:ec:5f brd ff:ff:ff:ff:ff:ff
   vf 0 MAC 00:00:00:00:00:00, link-state enable
   vf 1 MAC 00:00:00:00:00:00, link-state enable
   vf 2 MAC 00:00:00:00:00:00, link-state enable
   vf 3 MAC 00:00:00:00:00:00, link-state enable
   vf 4 MAC 00:00:00:00:00:00, link-state enable
   vf 5 MAC 00:00:00:00:00:00, link-state enable
   vf 6 MAC 00:00:00:00:00:00, link-state enable
3: eth1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP mode DEFAULT group default qlen 1000
   link/ether fa:16:3e:29:25:45 brd ff:ff:ff:ff:ff:ff
   vf 0 MAC 00:00:00:00:00:00, link-state enable
   vf 1 MAC 00:00:00:00:00:00, link-state enable
   vf 2 MAC 00:00:00:00:00:00, link-state enable
   vf 3 MAC 00:00:00:00:00:00, link-state enable
   vf 4 MAC 00:00:00:00:00:00, link-state enable
   vf 5 MAC 00:00:00:00:00:00, link-state enable
   vf 6 MAC 00:00:00:00:00:00, link-state enable
4: eth2: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN mode DEFAULT group default qlen 1000
   link/ether 48:8e:ef:78:74:59 brd ff:ff:ff:ff:ff:ff
5: eth3: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN mode DEFAULT group default qlen 1000
   link/ether 48:8e:ef:78:74:5a brd ff:ff:ff:ff:ff:ff
6: eth4: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN mode DEFAULT group default qlen 1000
   link/ether 48:8e:ef:78:74:5b brd ff:ff:ff:ff:ff:ff
7: eth5: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN mode DEFAULT group default qlen 1000
   link/ether 48:8e:ef:78:74:5c brd ff:ff:ff:ff:ff:ff
8: eth6: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN mode DEFAULT group default qlen 1000
   link/ether 00:2e:c7:e0:b2:37 brd ff:ff:ff:ff:ff:ff
9: eth7: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN mode DEFAULT group default qlen 1000
   link/ether 00:2e:c7:e0:b2:38 brd ff:ff:ff:ff:ff:ff
```

说明

其中，MAC 地址为“fa:16”开头的网卡为承载 VPC 网络的网络设备，如 eth0, eth1；MAC 地址为[增强高速网卡查看方式](#)中显示的 MAC 地址的网卡为承载增强高速网络的网络设备，如 eth6, eth7。

步骤 3 修改网卡配置文件

执行以下命令，编辑“/etc/sysconfig/network-scripts/ifcfg-eth6”和“/etc/sysconfig/network-scripts/ifcfg-eth7”，配置“eth6”设备和“eth7”设备的网络配置文件。

- **vi /etc/sysconfig/network-scripts/ifcfg-eth6**

“eth6”按以下格式编辑：

```
USERCTL=no
MTU=8888
NM_CONTROLLED=no
BOOTPROTO=static
DEVICE=eth6
TYPE=Ethernet
ONBOOT=yes
MASTER=bond1
SLAVE=yes
```

其中，“BOOTPROTO”参数取值修改为“static”，“DEVICE”为对应的网络设备名称，取值即为“eth6”，“MASTER”为对应的增强高速网卡 bond 的名称，取值如“bond1”，其他参数可保持不变。

- **vi /etc/sysconfig/network-scripts/ifcfg-eth7**

“eth7”按以下格式编辑（格式和规则和“eth6”一致）：

```
USERCTL=no
MTU=8888
NM_CONTROLLED=no
BOOTPROTO=static
DEVICE=eth7
```

```
TYPE=Ethernet
ONBOOT=yes
MASTER=bond1
SLAVE=yes
```

步骤4 修改 bond 配置文件

执行以下命令，编辑“/etc/sysconfig/network-scripts/ifcfg-bond1”。

```
vi /etc/sysconfig/network-scripts/ifcfg-bond1
```

按以下格式编辑：

```
MACADDR=00:2e:c7:e0:b2:37
BONDING MASTER=yes
USERCTL=no
ONBOOT=yes
NM CONTROLLED=no
BOOTPROTO=static
BONDING OPTS="mode=1 miimon=100"
DEVICE=bond1
TYPE=Bond
IPADDR=10.10.10.101
NETMASK=255.255.255.0
MTU=8888
```

其中，

- “MACADDR” 参数取值修改为增强高速网卡“eth6”或者“eth7”设备的 MAC 地址。
- “BOOTPROTO” 参数取值修改为“static”。
- “DEVICE” 参数取值修改为“bond1”。
- “IPADDR” 参数取值修改为待给“bond1”分配的 IP 地址（为增强高速网络规划的 IP 地址在没有与 VPC 网段冲突的情况下可任意规划，需要通过增强高速网络通信的物理机须将增强高速网络配置在同一个网段），如“10.10.10.101”。
- “NETMASK” 参数为给增强高速网络“bond1”配置的 IP 的子网掩码。

其他参数可保持不变。

修改完成后，按“Esc”，输入:wq 保存并退出。

步骤5 启动增强高速网络端口

执行以下命令，启动增强高速网络端口组“bond1”。

首先启动增强高速网卡“eth6”和“eth7”设备。

```
ifup eth6
```

```
ifup eth7
```

```
ifup bond1
```

```
[root@bms-centos network-scripts]# ifup bond1
Determining if ip address 10.10.10.101 is already in use for device bond1...
```

步骤 6 参见上述步骤，完成其他物理机的配置。

步骤 7 待其他物理机配置完成后，互相 ping 对端增强高速网络配置的同网段 IP，检查是否可以 ping 通。

```
[root@bms-centos network-scripts]# ping 10.10.10.102 -I bond1
PING 10.10.10.102 (10.10.10.102) from 10.10.10.101 bond1: 56(84) bytes of data.
 64 bytes from 10.10.10.102: icmp_seq=1 ttl=64 time=0.475 ms
 64 bytes from 10.10.10.102: icmp_seq=2 ttl=64 time=0.033 ms
 64 bytes from 10.10.10.102: icmp_seq=3 ttl=64 time=0.032 ms
^C
--- 10.10.10.102 ping statistics ---
 3 packets transmitted, 3 received, 0% packet loss, time 2263ms
 rtt min/avg/max/mdev = 0.032/0.180/0.475/0.208 ms
```

---结束

如果需要配置 VLAN，则按照以下步骤配置。

步骤 1 根据需要配置的 VLAN，配置相应 VLAN 子接口。假设 VLAN 为 316，执行以下命令，编辑“/etc/sysconfig/network-scripts/ifcfg-bond1.316”。

```
vi /etc/sysconfig/network-scripts/ifcfg-bond1.316
```

按以下格式编辑：

```
USERCTL=no
ONBOOT=yes
NM_CONTROLLED=no
BOOTPROTO=static
DEVICE=bond1.316
TYPE=Ethernet
IPADDR=10.10.0.101
NETMASK=255.255.255.0
VLAN=yes
PHYSDEV=bond1
```

其中，

- “DEVICE” 参数取值修改为新增的 bond 子接口名称。
- “IPADDR” 参数取值修改为待给“bond1.316”分配的 IP 地址（为增强高速网卡 VLAN 子接口规划的 IP 地址在没有与 VPC 网段冲突的情况下可任意规划，需要通过增强高速网卡 VLAN 子接口通信的物理机须将增强高速网卡 VLAN 子接口配置在同一个网段），如“10.10.0.101”。
- “NETMASK” 参数为给增强高速网卡“bond1.316”配置的 IP 的子网掩码。

其他参数可保持不变。

修改完成后，按“Esc”，输入:wq 保存并退出。

步骤 2 待其他物理机配置完成后，互相 ping 对端增强高速网络 VLAN 子接口的同网段 IP，检查是否可以 ping 通。

```
[root@bms-centos ~]# ping 10.10.0.102 -I bond1.316
PING 10.10.0.102 (10.10.0.102) from 10.10.0.101 bond1.316: 56(84) bytes of data.
64 bytes from 10.10.0.102: icmp_seq=1 ttl=64 time=0.681 ms
64 bytes from 10.10.0.102: icmp_seq=2 ttl=64 time=0.035 ms
64 bytes from 10.10.0.102: icmp_seq=3 ttl=64 time=0.031 ms
64 bytes from 10.10.0.102: icmp_seq=4 ttl=64 time=0.030 ms
^C
--- 10.10.0.102 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3342ms
rtt min/avg/max/mdev = 0.030/0.194/0.681/0.281 ms
```

---结束

删除网卡:

1. 获取待删除增强高速网卡的 bond 网卡地址。
2. 以“root”用户，使用密钥或密码登录物理机。
3. 找到 bond 关联的网络设备，然后执行命令关闭并删除网络设备。如果 bond 有 VLAN 子接口，这一步将自动删除 VLAN 子接口。

```
[root@bms-centos ~]# ifdown eth2
[root@bms-centos ~]# ifdown eth3
[root@bms-centos ~]# ifdown bond1
[root@bms-centos ~]# ip link delete bond1
[root@bms-centos ~]# ip link
1: lo: <LOOPBACK,UP,LOWER UP> mtu 65536 qdisc noqueue state UNKNOWN
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
2: eth0: <BROADCAST,MULTICAST,SLAVE,UP,LOWER UP> mtu 8888 qdisc mq master
bond0 state UP qlen 1000
    link/ether fa:16:00:6d:80:29 brd ff:ff:ff:ff:ff:ff
3: eth1: <BROADCAST,MULTICAST,SLAVE,UP,LOWER UP> mtu 8888 qdisc mq master
bond0 state UP qlen 1000
    link/ether fa:16:00:6d:80:29 brd ff:ff:ff:ff:ff:ff
4: eth2: <BROADCAST,MULTICAST> mtu 8888 qdisc mq state DOWN qlen 1000
    link/ether 40:7d:0f:52:e3:a5 brd ff:ff:ff:ff:ff:ff
5: eth3: <BROADCAST,MULTICAST> mtu 8888 qdisc mq state DOWN qlen 1000
    link/ether 40:7d:0f:52:e3:a6 brd ff:ff:ff:ff:ff:ff
6: bond0: <BROADCAST,MULTICAST,PROMISC,MASTER,UP,LOWER UP> mtu 8888 qdisc
noqueue state UP
    link/ether fa:16:00:6d:80:29 brd ff:ff:ff:ff:ff:ff
```

4. 执行以下命令，删除网络配置文件“/etc/sysconfig/network-scripts/ifcfg-eth2”、“/etc/sysconfig/network-scripts/ifcfg-eth3”和“/etc/sysconfig/network-scripts/ifcfg-bond1”。

```
rm -f /etc/sysconfig/network-scripts/ifcfg-eth2
```

```
rm -f /etc/sysconfig/network-scripts/ifcfg-eth3
```

```
rm -f /etc/sysconfig/network-scripts/ifcfg-bond1
```

如果有 VLAN 子接口，删除网络配置文件“/etc/sysconfig/network-scripts/ifcfg-bond1.vlan”，其中 vlan 表示 VLAN 子接口的 VLAN，如 316。

```
rm -f /etc/sysconfig/network-scripts/ifcfg-bond1.316
```

4.11.7 配置增强高速网卡 (Ubuntu 系列)

下面以 Ubuntu 16.04 LTS (Xenial Xerus x86_64)操作系统为例，举例介绍物理机增强高速网卡

的 bond 配置方法:

📖 说明

Ubuntu 系列其他操作系统的配置方法与 Ubuntu 16.04 LTS (Xenial Xerus x86_64)类似。

增加网卡:

步骤 1 以“root”用户，使用密钥或密码登录物理机。

步骤 2 进入物理机的命令行界面，查询网卡信息。

ip link

返回信息示例如下:

```
1: lo: <LOOPBACK,UP,LOWER UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1
    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,SLAVE,UP,LOWER UP> mtu 8888 qdisc mq master bond0
state UP group default qlen 1000
    link/ether fa:16:00:9b:91:c3 brd ff:ff:ff:ff:ff:ff
3: eth1: <BROADCAST,MULTICAST,SLAVE,UP,LOWER UP> mtu 8888 qdisc mq master bond0
state UP group default qlen 1000
    link/ether fa:16:00:9b:91:c3 brd ff:ff:ff:ff:ff:ff
4: p5p1: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN group default qlen 1000
    link/ether 40:7d:0f:52:e4:1d brd ff:ff:ff:ff:ff:ff
5: p5p2: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN group default qlen 1000
    link/ether 40:7d:0f:52:e4:1e brd ff:ff:ff:ff:ff:ff
6: p4p1: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN group default qlen 1000
    link/ether 40:7d:0f:52:e3:a9 brd ff:ff:ff:ff:ff:ff
7: p4p2: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN group default qlen 1000
    link/ether 40:7d:0f:52:e3:aa brd ff:ff:ff:ff:ff:ff
8: bond0: <BROADCAST,MULTICAST,MASTER,UP,LOWER_UP> mtu 8888 qdisc noqueue state UP
group default qlen 1000
    link/ether fa:16:00:9b:91:c3 brd ff:ff:ff:ff:ff:ff
    inet 192.168.254.85/24 brd 192.168.254.255 scope global bond0
        valid_lft forever preferred_lft forever
    inet6 fe80::f816:ff:fe9b:91c3/64 scope link
        valid_lft forever preferred_lft forever
9: bond0.3157@bond0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 8888 qdisc noqueue
state UP group default qlen 1000
    link/ether fa:16:00:9c:1e:79 brd ff:ff:ff:ff:ff:ff
    inet 192.168.100.14/24 brd 192.168.100.255 scope global bond0.3157
        valid_lft forever preferred_lft forever
```

```
inet6 fe80::f816:ff:fe9c:1e79/64 scope link
    valid_lft forever preferred_lft forever
10: bond0.3159@bond0: <BROADCAST,MULTICAST,UP,LOWER UP> mtu 8888 qdisc noqueue
state UP group default qlen 1000
    link/ether fa:16:00:0a:2e:8e brd ff:ff:ff:ff:ff:ff
    inet 192.168.101.153/24 brd 192.168.101.255 scope global bond0.3159
        valid_lft forever preferred_lft forever
    inet6 fe80::f816:ff:fe0a:2e8e/64 scope link
        valid_lft forever preferred_lft forever
```

📖 说明

其中，“eth0”和“eth1”为承载VPC网络的网络设备，“p5p1”、“p5p2”、“p4p1”和“p4p2”为承载增强高速网络的网络设备。下面步骤将使用“p4p1”和“p4p2”配置增强高速网卡 bond。

步骤3 检查 udev 规则配置文件

执行以下命令，查看“/etc/udev/rules.d/”目录下是否有“80-persistent-net.rules”配置文件。

```
ll /etc/udev/rules.d/ | grep 80-persistent-net.rules
```

- 如果存在“80-persistent-net.rules”，且该配置文件中已存在[步骤2](#)中查询到的除“bond0”和“lo”以外的其它所有网卡和对应的MAC地址，请执行[步骤6](#)。
- 否则，继续执行[步骤4](#)。

步骤4 执行以下命令，将“/etc/udev/rules.d/70-persistent-net.rules”文件拷贝一份（文件名为“/etc/udev/rules.d/80-persistent-net.rules”）。

```
cp -p /etc/udev/rules.d/70-persistent-net.rules /etc/udev/rules.d/80-persistent-net.rules
```

步骤5 设置 udev 规则。

将[步骤2](#)中查询到的除“lo”、“eth0”、“eth1”和“bond0”以外的网卡和MAC对应关系添加到“/etc/udev/rules.d/80-persistent-net.rules”文件中，使得物理机重启复位后，网卡名称和顺序不会发生改变。

📖 说明

网卡的MAC地址和名称中的字母，请使用小写字母。

```
vim /etc/udev/rules.d/80-persistent-net.rules
```

修改后的示例如下：

```
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="*", ATTR{address}=="f4:4c:7f:5d:b6:fc",
NAME="eth0"
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="*", ATTR{address}=="f4:4c:7f:5d:b6:fd",
NAME="eth1"
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="*", ATTR{address}=="40:7d:0f:52:e4:1d",
NAME="p5p1"
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="*", ATTR{address}=="40:7d:0f:52:e4:1e",
NAME="p5p2"
```

```
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="?* ", ATTR{address}=="40:7d:0f:52:e3:a9",  
NAME="p4p1"  
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="?* ", ATTR{address}=="40:7d:0f:52:e3:aa",  
NAME="p4p2"
```

修改完成后，按“Esc”，输入:wq 保存并退出。

步骤 6 执行以下命令，将网卡配置文件“/etc/network/interfaces.d/50-cloud-init.cfg”拷贝为“/etc/network/interfaces.d/60-cloud-init.cfg”。

```
cp -p /etc/network/interfaces.d/50-cloud-init.cfg /etc/network/interfaces.d/60-cloud-init.cfg
```

说明

如果没有“/etc/network/interfaces.d/50-cloud-init.cfg”文件，请拷贝“/etc/network/interfaces”文件，依次执行以下命令：

```
mkdir /etc/network/interfaces.d
```

```
cp -p /etc/network/interfaces /etc/network/interfaces.d/60-cloud-init.cfg
```

步骤 7 执行以下命令，编辑“/etc/network/interfaces.d/60-cloud-init.cfg”，配置“p4p1”设备和“p4p2”设备的网络配置文件“/etc/network/interfaces.d/60-cloud-init.cfg”。

```
vim /etc/network/interfaces.d/60-cloud-init.cfg
```

按如下格式编辑：

```
auto p4p1  
iface p4p1 inet manual  
bond mode 1  
bond-master bond1  
bond miimon 100  
mtu 8888  
  
auto p4p2  
iface p4p2 inet manual  
bond mode 1  
bond-master bond1  
bond miimon 100  
mtu 8888  
  
auto bond1  
iface bond1 inet static  
bond miimon 100  
bond-slaves none  
bond mode 1  
address 10.10.10.103  
netmask 255.255.255.0  
hwaddress 40:7d:0f:52:e3:a9  
mtu 8888
```

其中，

- “p4p1”和“p4p2”为承载增强高速网络配置的网卡名称。
- “hwaddress”为“p4p1”设备对应的 MAC 地址。

- “address”的取值为给增强高速网络“bond1”配置的IP（给增强高速网络规划的IP地址在没有与VPC网段冲突的情况下可任意规划，需要通过增强高速网络通信的物理机须将增强高速网络配置在同一个网段）。
- “netmask”的取值为给增强高速网络“bond1”配置的IP的掩码。

各个设备的其他参数可参考如上信息进行配置，如“mtu”配置为“8888”，“bond_miimon”配置为“100”，“bond_mode”配置为“1”等。

修改完成后，按“Esc”，输入:wq保存并退出。

步骤 8 执行以下命令，启动 bond 网卡。

```
ifup p4p1
```

```
ifup p4p2
```

📖 说明

其中，“p4p1”和“p4p2”分别为承载增强高速网络的网卡。

步骤 9 执行以下命令，查看网卡设备的状态和“bond1”配置文件是否生效。

ip link

```
root@bms-ubuntu:~# ip link
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN mode DEFAULT group default qlen 1
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
2: eth0: <BROADCAST,MULTICAST,SLAVE,UP,LOWER_UP> mtu 8888 qdisc mq master bond0 state UP mode DEFAULT group default qlen 1000
    link/ether fa:16:00:9b:91:c3 brd ff:ff:ff:ff:ff:ff
3: eth1: <BROADCAST,MULTICAST,SLAVE,UP,LOWER_UP> mtu 8888 qdisc mq master bond0 state UP mode DEFAULT group default qlen 1000
    link/ether fa:16:00:9b:91:c3 brd ff:ff:ff:ff:ff:ff
4: p5p1: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN mode DEFAULT group default qlen 1000
    link/ether 40:7d:0f:52:e4:1d brd ff:ff:ff:ff:ff:ff
5: p5p2: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN mode DEFAULT group default qlen 1000
    link/ether 40:7d:0f:52:e4:1e brd ff:ff:ff:ff:ff:ff
6: p4p1: <BROADCAST,MULTICAST,SLAVE,UP,LOWER_UP> mtu 8888 qdisc mq master bond1 state UP mode DEFAULT group default qlen 1000
    link/ether 40:7d:0f:52:e3:a9 brd ff:ff:ff:ff:ff:ff
7: p4p2: <BROADCAST,MULTICAST,SLAVE,UP,LOWER_UP> mtu 8888 qdisc mq master bond1 state UP mode DEFAULT group default qlen 1000
    link/ether 40:7d:0f:52:e3:a9 brd ff:ff:ff:ff:ff:ff
8: bond0: <BROADCAST,MULTICAST,MASTER,UP,LOWER_UP> mtu 8888 qdisc noqueue state UP mode DEFAULT group default qlen 1000
    link/ether fa:16:00:9b:91:c3 brd ff:ff:ff:ff:ff:ff
12: bond1: <BROADCAST,MULTICAST,MASTER,UP,LOWER_UP> mtu 8888 qdisc noqueue state UP mode DEFAULT group default qlen 1000
    link/ether 40:7d:0f:52:e3:a9 brd ff:ff:ff:ff:ff:ff
13: bond1.316@bond1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 8888 qdisc noqueue state UP mode DEFAULT group default qlen 1000
    link/ether 40:7d:0f:52:e3:a9 brd ff:ff:ff:ff:ff:ff
```

lscfg

```

root@bms-ubuntu:~# ifconfig
bond0    Link encap:Ethernet  HWaddr fa:16:00:9b:91:c3
         inet addr:192.168.254.85  Bcast:192.168.254.255  Mask:255.255.255.0
         inet6 addr: fe80::f816:ff:fe9b:91c3/64 Scope:Link
         UP BROADCAST RUNNING MASTER MULTICAST  MTU:8888  Metric:1
         RX packets:6079 errors:0 dropped:1410 overruns:0 frame:0
         TX packets:3470 errors:0 dropped:0 overruns:0 carrier:0
         collisions:0 txqueuelen:1000
         RX bytes:1241961 (1.2 MB)  TX bytes:801316 (801.3 KB)

bond1    Link encap:Ethernet  HWaddr 40:7d:0f:52:e3:a9
         inet addr:10.10.10.103  Bcast:10.10.10.255  Mask:255.255.255.0
         inet6 addr: fe80::427d:fff:fe52:e3a9/64 Scope:Link
         UP BROADCAST RUNNING MASTER MULTICAST  MTU:8888  Metric:1
         RX packets:1285 errors:0 dropped:642 overruns:0 frame:0
         TX packets:707 errors:0 dropped:0 overruns:0 carrier:0
         collisions:0 txqueuelen:1000
         RX bytes:78202 (78.2 KB)  TX bytes:32534 (32.5 KB)

bond1.316 Link encap:Ethernet  HWaddr 40:7d:0f:52:e3:a9
         inet addr:10.10.0.103  Bcast:10.10.0.255  Mask:255.255.255.0
         inet6 addr: fe80::427d:fff:fe52:e3a9/64 Scope:Link
         UP BROADCAST RUNNING MULTICAST  MTU:8888  Metric:1
         RX packets:37 errors:0 dropped:0 overruns:0 frame:0
         TX packets:55 errors:0 dropped:0 overruns:0 carrier:0
         collisions:0 txqueuelen:1000
         RX bytes:2804 (2.8 KB)  TX bytes:4290 (4.2 KB)

eth0     Link encap:Ethernet  HWaddr fa:16:00:9b:91:c3
         UP BROADCAST RUNNING SLAVE MULTICAST  MTU:8888  Metric:1
         RX packets:1443 errors:0 dropped:1410 overruns:0 frame:0
         TX packets:715 errors:0 dropped:0 overruns:0 carrier:0
         collisions:0 txqueuelen:1000
         RX bytes:359890 (359.8 KB)  TX bytes:242442 (242.4 KB)

eth1     Link encap:Ethernet  HWaddr fa:16:00:9b:91:c3
         UP BROADCAST RUNNING SLAVE MULTICAST  MTU:8888  Metric:1
         RX packets:4669 errors:0 dropped:0 overruns:0 frame:0
         TX packets:2788 errors:0 dropped:0 overruns:0 carrier:0
         collisions:0 txqueuelen:1000
         RX bytes:892139 (892.1 KB)  TX bytes:568072 (568.0 KB)

lo       Link encap:Local Loopback
         inet addr:127.0.0.1  Mask:255.0.0.0
         inet6 addr: ::1/128 Scope:Host
         UP LOOPBACK RUNNING  MTU:65536  Metric:1
         RX packets:54 errors:0 dropped:0 overruns:0 frame:0
         TX packets:54 errors:0 dropped:0 overruns:0 carrier:0
         collisions:0 txqueuelen:1
         RX bytes:6048 (6.0 KB)  TX bytes:6048 (6.0 KB)

p4p1    Link encap:Ethernet  HWaddr 40:7d:0f:52:e3:a9
         UP BROADCAST RUNNING SLAVE MULTICAST  MTU:8888  Metric:1
         RX packets:643 errors:0 dropped:0 overruns:0 frame:0
         TX packets:738 errors:0 dropped:0 overruns:0 carrier:0
         collisions:0 txqueuelen:1000
         RX bytes:39682 (39.6 KB)  TX bytes:34192 (34.1 KB)

p4p2    Link encap:Ethernet  HWaddr 40:7d:0f:52:e3:a9
         UP BROADCAST RUNNING SLAVE MULTICAST  MTU:8888  Metric:1
         RX packets:663 errors:0 dropped:663 overruns:0 frame:0
         TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
         collisions:0 txqueuelen:1000
         RX bytes:39780 (39.7 KB)  TX bytes:0 (0.0 B)
    
```

步骤 10 参见上述步骤，完成其他物理机的配置。

步骤 11 待其他物理机配置完成后，互相 ping 对端增强高速网络配置的同网段 IP，检查是否可以 ping 通。

ping 10.10.10.102，返回如下：

```

[root@bms-ubuntu ~]# ping 10.10.10.102 -I bond1
PING 10.10.10.102 (10.10.10.102) from 10.10.10.103 bond1: 56(84) bytes of data.
64 bytes from 10.10.10.102: icmp_seq=1 ttl=64 time=0.681 ms
64 bytes from 10.10.10.102: icmp_seq=2 ttl=64 time=0.035 ms
64 bytes from 10.10.10.102: icmp_seq=3 ttl=64 time=0.031 ms
64 bytes from 10.10.10.102: icmp_seq=4 ttl=64 time=0.030 ms
    
```

```
^C
--- 10.10.10.102 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3342ms
```

---结束

如果需要配置 VLAN，则按照以下步骤配置。

步骤 1 根据需要配置的 VLAN，配置相应 VLAN 子接口。假设 VLAN 为 316，执行以下命令，编辑“/etc/network/interfaces.d/60-cloud-init.cfg”。

vim /etc/network/interfaces.d/60-cloud-init.cfg

按以下格式编辑：

```
auto p4p1
iface p4p1 inet manual
bond_mode 1
bond-master bond1
bond_miimon 100
mtu 8888

auto p4p2
iface p4p2 inet manual
bond_mode 1
bond-master bond1
bond_miimon 100
mtu 8888

auto bond1
iface bond1 inet static
bond miimon 100
bond-slaves none
bond mode 1
address 10.10.10.103
netmask 255.255.255.0
hwaddress 40:7d:0f:52:e3:a9
mtu 8888

auto bond1.316
iface bond1.316 inet static
bond miimon 100
bond-slaves none
bond mode 1
address 10.10.0.103
netmask 255.255.255.0
hwaddress 40:7d:0f:52:e3:a9
mtu 8888
```

步骤 2 执行以下命令，启动 bond 网卡 VLAN 子接口。

ifup bond1.316

步骤 3 待其他物理机配置完成后，互相 ping 对端增强高速网络 VLAN 子接口的同网段 IP，检查是否可以 ping 通。

```
root@bms-ubuntu:~# ping 10.10.0.102 -I bond1.316
PING 10.10.0.102 (10.10.0.102) from 10.10.0.103 bond1.316: 56(84) bytes of data.
64 bytes from 10.10.0.102: icmp_seq=1 ttl=64 time=0.053 ms
64 bytes from 10.10.0.102: icmp_seq=2 ttl=64 time=0.053 ms
64 bytes from 10.10.0.102: icmp_seq=3 ttl=64 time=0.047 ms
64 bytes from 10.10.0.102: icmp_seq=4 ttl=64 time=0.049 ms
64 bytes from 10.10.0.102: icmp_seq=5 ttl=64 time=0.046 ms
^C
--- 10.10.0.102 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 3996ms
rtt min/avg/max/mdev = 0.046/0.049/0.053/0.008 ms
```

----结束

删除网卡:

1. 获取待删除增强高速网卡的 bond 网卡地址。
2. 以“root”用户，使用密钥或密码登录物理机。
3. 找到 bond 关联的网络设备，然后执行命令关闭并删除网络设备。如果 bond 有 VLAN 子接口，这一步将自动删除 VLAN 子接口。

```
[root@bms-ubuntu ~]# ifdown p4p1
[root@bms-ubuntu ~]# ifdown p4p2
[root@bms-ubuntu ~]# ifdown bond1
```

4. 执行以下命令，删除网络配置文件“/etc/network/interfaces.d/60-cloud-init.cfg”。
- ```
rm -f /etc/network/interfaces.d/60-cloud-init.cfg
```

## 4.11.8 配置增强高速网卡（Windows Server 系列）

下面以 Windows Server 2012 R2 Standard 操作系统为例，举例介绍物理机增强高速网络 bond 的配置方法。

### 📖 说明

Windows Server 系列其他操作系统的配置方法与 Windows Server 2012 R2 Standard 类似。

## 增加网卡

**步骤 1** 登录 Windows 物理机。

**步骤 2** 进入物理机的 Windows PowerShell 命令行界面，执行以下命令，查询网卡信息。

### Get-NetAdapter

返回信息示例如下：

```
PS C:\Users\Administrator> Get-NetAdapter
Name InterfaceDescription ifIndex Status MacAddress
---- -
以太网 3 Intel(R) 82599 10 Gigabit 双端口网络... 15 Up 2C-55-D3-C4-
eth0_198befdc-4480-49...6 Intel(R) 82599 10 Gigabit 双端口网络... 14 Up 2C-55-D3-C4-
eth1_198befdc-4480-49...6 Intel(R) 82599 10 Gigabit 双端口网络... 17 Up 40-7D-0F-52-
以太网 4 Intel(R) 82599 10 Gigabit 双端口网络... 16 Disconnected 40-7D-0F-52-
Team1 Microsoft Network Adapter Multiplexo... 21 Up FA-16-00-86-
```

## 说明

其中，“eth0”和“eth1”为承载 VPC 网络的网络设备，“以太网 3”和“以太网 4”为承载增强高速网络 bond 的网络设备。下面步骤将使用“以太网 3”和“以太网 4”配置增强高速网络。

**步骤 3** 如果想提高 OS 侧出方向流量请参考[方法一](#)配置；如果对流量没有特殊要求，请参考[方法二](#)配置。

- **方法一：OS 内组 bond 为交换机独立模式，出方向流量可达到主主的效果，但入方向仍然与主备模式保持一致。**

1. 执行以下命令，创建增强高速网络 bond 的端口组。

```
New-NetLbfoTeam -Name qinq -TeamMembers "以太网 3","以太网 4" -
TeamingMode SwitchIndependent -LoadBalancingAlgorithm Dynamic -
Confirm:$false
```

```
PS C:\Users\Administrator> New-NetLbfoTeam -Name qinq -TeamMembers "以太网 3","以太网 4" -TeamingMode Switc
-LoadBalancingAlgorithm Dynamic -Confirm:$false
Name : qinq
Members : {以太网 4, 以太网 3}
TeamNics : qinq
TeamingMode : SwitchIndependent
LoadBalancingAlgorithm : Dynamic
Status : Degraded
```

## 说明

其中，“qinq”为给增强高速网络端口组规划的端口组名称，“以太网 3”和“以太网 4”为[步骤 2](#)中获取的承载增强高速网络的网络设备。

2. 执行以下命令，查询网络适配器列表。

```
get-NetLbfoTeamMember
```

```
PS C:\Users\Administrator> get-NetLbfoTeamMember

Name : eth0_d7a1277d-7cd9-4fd4-a1ff-a7c4d8009361
InterfaceDescription : Intel(R) Ethernet Connection X722 for 10GbE SFP+
Team : Team1
AdministrativeMode : Standby
OperationalStatus : Standby
TransmitLinkSpeed(Gbps) : 10
ReceiveLinkSpeed(Gbps) : 10
FailureReason : AdministrativeDecision

Name : eth1_d7a1277d-7cd9-4fd4-a1ff-a7c4d8009361
InterfaceDescription : Intel(R) Ethernet Connection X722 for 10GbE SFP+ #2
Team : Team1
AdministrativeMode : Active
OperationalStatus : Active
TransmitLinkSpeed(Gbps) : 10
ReceiveLinkSpeed(Gbps) : 10
FailureReason : NoFailure

Name : 以太网 4
InterfaceDescription : Intel(R) 82599 10 Gigabit 双端口网络连接 #2
Team : qing
AdministrativeMode : Active
OperationalStatus : Active
TransmitLinkSpeed(Gbps) : 10
ReceiveLinkSpeed(Gbps) : 10
FailureReason : NoFailure

Name : 以太网 3
InterfaceDescription : Intel(R) 82599 10 Gigabit 双端口网络连接
Team : qing
AdministrativeMode : Active
OperationalStatus : Active
TransmitLinkSpeed(Gbps) : 10
ReceiveLinkSpeed(Gbps) : 10
FailureReason : NoFailure
```

### Get-NetAdapter

```
PS C:\Users\Administrator> Get-NetAdapter

Name InterfaceDescription ifIndex Status MacAddress

qing Microsoft Network Adapter Multiple...#2 33 Up DC-99-14-93-DE-C2
eth1_d7a1277d-7...8009361 Intel(R) Ethernet Connection X722 ...#2 19 Up 2C-97-B1-D2-B4-87
LOM4 Intel(R) Ethernet Connection X722 fo... 17 Disconnected 2C-97-B1-D2-B4-89
Team1 Microsoft Network Adapter Multiplexo... 24 Up FA-16-3E-35-C9-F3
eth0_d7a1277d-7...8009361 Intel(R) Ethernet Connection X722 fo... 15 Up 2C-97-B1-D2-B4-86
LOM3 Intel(R) Ethernet Connection X722 ...#2 18 Disconnected 2C-97-B1-D2-B4-88
以太网 3 Intel(R) 82599 10 Gigabit 双端口网络... 14 Up DC-99-14-93-DE-C3
以太网 4 Intel(R) 82599 10 Gigabit 双端口网络... 16 Up DC-99-14-93-DE-C2
```

## • 方法二：主备模式 OS 内组 bond。

3.1 执行以下命令，创建增强高速网络 bond 的端口组。

**New-NetLbfoTeam -Name Team2 -TeamMembers "以太网 3","以太网 4" -TeamingMode SwitchIndependent -LoadBalancingAlgorithm IPAddresses -Confirm:\$false**

```
PS C:\Users\Administrator> New-NetLbfoTeam -Name Team2 -TeamMembers "以太网 3","以太网 4" -TeamingMode SwitchIndependent
-LoadBalancingAlgorithm IPAddresses -Confirm:$false

Name : Team2
Members : {以太网 4, 以太网 3}
TeamName : Team2
TeamingMode : SwitchIndependent
LoadBalancingAlgorithm : IPAddresses
Status : Degraded
```

### 📖 说明

其中，“Team2”为给增强高速网络端口组规划的端口组名称，“以太网 3”和“以太网 4”为[步骤 2](#)中获取的承载增强高速网络的网络设备。

3.2 执行以下命令，设置[步骤 3.1](#)中创建的“Team2”端口组中一个网口模式为备用模式。

**Set-NetLbfoTeamMember -Name "以太网 4" -AdministrativeMode Standby -Confirm:\$false**

### 📖 说明

当前增强高速网络配置的端口组只支持主备模式，其中，“以太网 4”为组成增强高速网络端口组中的其中一个端口，配置哪个端口为备用端口，根据您的规划自行决定。

### get-NetLbfoTeamMember

```
PS C:\Users\Administrator> get-NetLbfoTeamMember

Name : eth1_198befdc-4480-4999-a2ab-d910f4e0d8e6
InterfaceDescription : Intel(R) 82599 10 Gigabit 双端口网络连接 #4
Team : Team1
AdministrativeMode : Active
OperationalStatus : Active
TransmitLinkSpeed(Gbps) : 10
ReceiveLinkSpeed(Gbps) : 10
FailureReason : NoFailure

Name : eth0_198befdc-4480-4999-a2ab-d910f4e0d8e6
InterfaceDescription : Intel(R) 82599 10 Gigabit 双端口网络连接
Team : Team1
AdministrativeMode : Standby
OperationalStatus : Standby
TransmitLinkSpeed(Gbps) : 10
ReceiveLinkSpeed(Gbps) : 10
FailureReason : AdministrativeDecision

Name : 以太网 4
InterfaceDescription : Intel(R) 82599 10 Gigabit 双端口网络连接 #3
Team : Team2
AdministrativeMode : Standby
OperationalStatus : Failed
TransmitLinkSpeed(Mbps) : 0
ReceiveLinkSpeed(Mbps) : 0
FailureReason : PhysicalMediaDisconnected

Name : 以太网 3
InterfaceDescription : Intel(R) 82599 10 Gigabit 双端口网络连接 #2
Team : Team2
AdministrativeMode : Active
OperationalStatus : Active
TransmitLinkSpeed(Gbps) : 10
ReceiveLinkSpeed(Gbps) : 10
FailureReason : NoFailure
```

### Get-NetAdapter

```
PS C:\Users\Administrator> Get-NetAdapter

Name InterfaceDescription ifIndex Status MacAddress LinkSpeed

Team2 Microsoft Network Adapter Multiplexe... 30 Up 40-7D-0F-52-E3-AD 10 Gbps
以太网 3 Intel(R) 82599 10 Gigabit 双端口网络... 15 Up 2C-55-D3-C4-9C-5A 10 Gbps
eth0_198befdc-4480-49...6 Intel(R) 82599 10 Gigabit 双端口网络... 14 Up 2C-55-D3-C4-9C-59 10 Gbps
eth1_198befdc-4480-49...6 Intel(R) 82599 10 Gigabit 双端口网络... 17 Up 40-7D-0F-52-E3-AE 10 Gbps
以太网 4 Intel(R) 82599 10 Gigabit 双端口网络... 16 Disconnected 40-7D-0F-52-E3-AD 0 bps
Team1 Microsoft Network Adapter Multiplexo... 21 Up FA-16-00-86-98-83 10 Gbps
```

**步骤 4** 执行以下命令，进入“网络连接”界面。

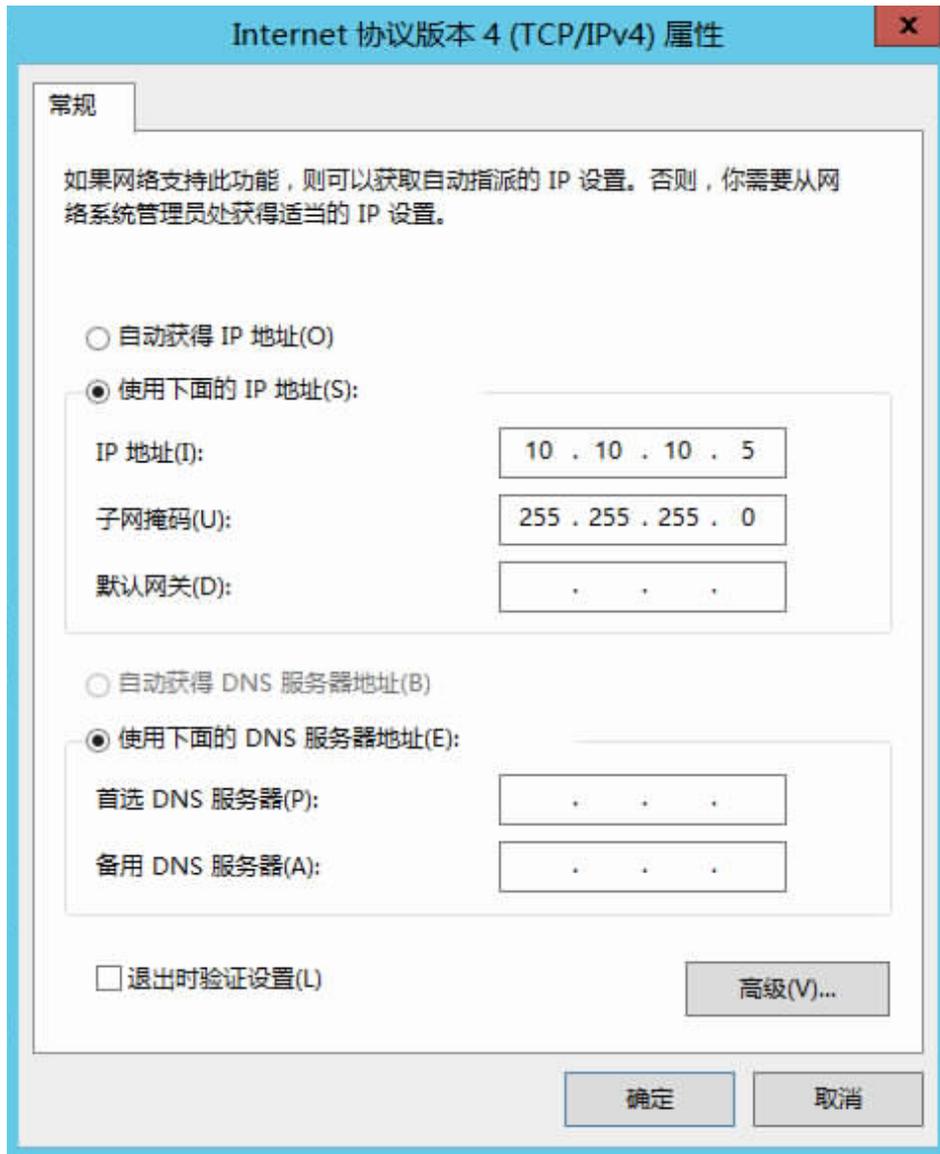
`ncpa.cpl`

执行完成后，进入如下界面：



**步骤 5** 配置增强高速网络。

1. 在“网络连接”界面，双击在[步骤 3](#)中创建的端口组“Team2”，进入“Team2 状态”页面。
2. 单击“属性”，进入“Team2 属性”页面。
3. 在“网络”页签下双击“Internet 协议版本 4 (TCP/IPv4)”，进入“Internet 协议版本 4 (TCP/IPv4) 属性”页面。
4. 选择“使用下面的 IP 地址”，配置增强高速网络的 IP 地址和子网掩码，单击“确定”。



#### 说明

其中，为增强高速网络规划的 IP 地址在没有与 VPC 网段冲突的情况下可任意规划，需要通过增强高速网络通信的物理机须将增强高速网络配置在同一个网段。

**步骤 6** 参见上述步骤，完成其他物理机的配置。

**步骤 7** 待其他物理机配置完成后，互相 ping 对端增强高速网络配置的同网段 IP，检查是否可以 ping 通。

```

PS C:\Users\Administrator> ping 10.10.10.4

正在 Ping 10.10.10.4 具有 32 字节的数据:
来自 10.10.10.4 的回复: 字节=32 时间<1ms TTL=128

10.10.10.4 的 Ping 统计信息:
 数据包: 已发送 = 4, 已接收 = 4, 丢失 = 0 (0% 丢失),
 往返行程的估计时间(以毫秒为单位):
 最短 = 0ms, 最长 = 0ms, 平均 = 0ms
PS C:\Users\Administrator>

```

---结束

## 删除网卡

1. 登录 Windows 物理机。
2. 进入物理机的 Windows PowerShell 命令行界面，执行以下命令，查询需要删除的增强高速网卡 bond 信息。

### Get-NetLbfoTeamNIC -Team Team2

```

PS C:\Users\Administrator> Get-NetLbfoTeamNIC -Team Team2

Name : Team2
InterfaceDescription : Microsoft Network Adapter Multiplexor Driver #2
Team : Team2
VlanID :
Primary : True
Default : True
TransmitLinkSpeed(Gbps) : 10
ReceiveLinkSpeed(Gbps) : 10

```

3. 执行以下命令，删除网卡 bond。

### Remove-NetLbfoTeam -Name "Team2"

```

PS C:\Users\Administrator> Remove-NetLbfoTeam -Name Team2

确认
是否确实要执行此操作?
Removes the Team: 'Team2' from the system.
[Y] 是(Y) [A] 全是(A) [N] 否(N) [L] 全否(L) [S] 挂起(S) [?] 帮助 (默认值为“Y”): Y
PS C:\Users\Administrator>

```

4. 执行如下命令，查询网络信息，确认网卡已被删除。

### Get-NetAdapter

```

PS C:\Users\Administrator> Get-NetAdapter

Name InterfaceDescription ifIndex Status MacAddress LinkSpeed
----- -
以太网 3 Intel(R) 82599 10 Gigabit 双端口网络... 15 Up 2C-55-D3-C4-9C-5A 10 Gbps
eth0_198befdc-4480-49...6 Intel(R) 82599 10 Gigabit 双端口网络... 14 Up 2C-55-D3-C4-9C-59 10 Gbps
eth1_198befdc-4480-49...6 Intel(R) 82599 10 Gigabit 双端口网络... 17 Up 40-7D-0F-52-E3-AE 10 Gbps
以太网 4 Intel(R) 82599 10 Gigabit 双端口网络... 16 Up 40-7D-0F-52-E3-AD 10 Gbps
Team1 Microsoft Network Adapter Multiplexo... 21 Up FA-16-00-86-9B-83 10 Gbps

```

## 4.12 自定义 VLAN 网络配置

### 4.12.1 SUSE Linux Enterprise Server 12 系列

本章节主要介绍 SUSE Linux Enterprise Server 12 系列自定义 VLAN 网络配置。

#### 注意事项

自定义 VLAN 网络网段不能与现有的物理机上已经配置的网段重叠。

下面以 SUSE Linux Enterprise Server 12 SP1 (x86\_64) 操作系统为例，举例介绍物理机的自定义 VLAN 网络配置方法：

**步骤 1** 以“root”用户，使用密钥或密码登录物理机。

**步骤 2** 进入物理机的命令行界面，查询网卡信息。

```
ip link
```

返回信息示例如下：

```
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN mode
DEFAULT group default
 link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
2: eth0: <BROADCAST,MULTICAST,SLAVE,UP,LOWER_UP> mtu 8888 qdisc mq master bond0
state UP mode DEFAULT group default qlen 1000
 link/ether fa:16:3e:3d:1c:e0 brd ff:ff:ff:ff:ff:ff
3: eth1: <BROADCAST,MULTICAST,SLAVE,UP,LOWER_UP> mtu 8888 qdisc mq master bond0
state UP mode DEFAULT group default qlen 1000
 link/ether fa:16:3e:3d:1c:e0 brd ff:ff:ff:ff:ff:ff
4: eth2: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state DOWN mode
DEFAULT group default qlen 1000
 link/ether 38:4c:4f:89:55:8d brd ff:ff:ff:ff:ff:ff
5: eth3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state DOWN mode
DEFAULT group default qlen 1000
 link/ether 38:4c:4f:89:55:8e brd ff:ff:ff:ff:ff:ff
6: bond0: <BROADCAST,MULTICAST,MASTER,UP,LOWER_UP> mtu 8888 qdisc noqueue state
UP mode DEFAULT group default
```

```

link/ether fa:16:3e:3d:1c:e0 brd ff:ff:ff:ff:ff:ff
7: bond0.3133@bond0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 8888 qdisc noqueue
state UP mode DEFAULT group default
link/ether fa:16:3e:57:87:6e brd ff:ff:ff:ff:ff:ff

```

### 说明

其中，“eth0”和“eth1”为承载 VPC 网络的网络设备，“eth2”和“eth3”为承载自定义 VLAN 网络的网络设备。

### 步骤 3 设置 udev 规则。

执行以下命令创建“80-persistent-net.rules”文件。

```
cp /etc/udev/rules.d/70-persistent-net.rules /etc/udev/rules.d/80-persistent-net.rules
```

将[步骤 2](#)中查询到的，且“80-persistent-net.rules”中未体现的网卡 MAC 地址和名称，写入该文件中，使得物理机重启复位后，网卡名称和顺序不会发生改变。

说明

网卡的 MAC 地址和名称中的字母，请使用小写字母。

```
vim /etc/udev/rules.d/80-persistent-net.rules
```

修改后的示例如下：

```

SUBSYSTEM=="net", ACTION=="add", DRIVERS=="?*",
ATTR{address}=="38:4c:4f:29:0b:e0", NAME="eth0"
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="?*",
ATTR{address}=="38:4c:4f:29:0b:e1", NAME="eth1"
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="?*",
ATTR{address}=="38:4c:4f:89:55:8d", NAME="eth2"
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="?*",
ATTR{address}=="38:4c:4f:89:55:8e", NAME="eth3"

```

修改完成后，保存并退出。

### 步骤 4 查询网卡的 IP 信息。

```
ifconfig
```

返回信息示例如下，其中的“bond0”和“bond0.313”为申请物理机时自动分配的网卡平面 IP 地址。

```

bond0 Link encap:Ethernet HWaddr FA:16:3E:3D:1C:E0
 inet addr:10.0.1.2 Bcast:10.0.1.255 Mask:255.255.255.0
 inet6 addr: fe80::f816:3eff:fe3d:1ce0/64 Scope:Link

```

```
UP BROADCAST RUNNING MASTER MULTICAST MTU:8888 Metric:1
RX packets:852 errors:0 dropped:160 overruns:0 frame:0
TX packets:1121 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:0
RX bytes:125429 (122.4 Kb) TX bytes:107221 (104.7 Kb)
```

```
bond0.313 Link encap:Ethernet HWaddr FA:16:3E:57:87:6E
 inet addr:10.0.3.2 Bcast:10.0.3.255 Mask:255.255.255.0
 inet6 addr: fe80::f816:3eff:fe57:876e/64 Scope:Link
UP BROADCAST RUNNING MULTICAST MTU:8888 Metric:1
RX packets:169 errors:0 dropped:0 overruns:0 frame:0
TX packets:13 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:0
RX bytes:8684 (8.4 Kb) TX bytes:1696 (1.6 Kb)
```

```
eth0 Link encap:Ethernet HWaddr FA:16:3E:3D:1C:E0
 UP BROADCAST RUNNING SLAVE MULTICAST MTU:8888 Metric:1
 RX packets:428 errors:0 dropped:10 overruns:0 frame:0
 TX packets:547 errors:0 dropped:0 overruns:0 carrier:0
 collisions:0 txqueuelen:1000
 RX bytes:64670 (63.1 Kb) TX bytes:50132 (48.9 Kb)
```

```
eth1 Link encap:Ethernet HWaddr FA:16:3E:3D:1C:E0
 UP BROADCAST RUNNING SLAVE MULTICAST MTU:8888 Metric:1
 RX packets:424 errors:0 dropped:7 overruns:0 frame:0
 TX packets:574 errors:0 dropped:0 overruns:0 carrier:0
 collisions:0 txqueuelen:1000
 RX bytes:60759 (59.3 Kb) TX bytes:57089 (55.7 Kb)
```

```
lo Link encap:Local Loopback
 inet addr:127.0.0.1 Mask:255.0.0.0
 inet6 addr: ::1/128 Scope:Host
 UP LOOPBACK RUNNING MTU:65536 Metric:1
 RX packets:8 errors:0 dropped:0 overruns:0 frame:0
```

```
TX packets:8 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:0
RX bytes:520 (520.0 b) TX bytes:520 (520.0 b)
```

### 步骤 5 查询组成 bond 的网卡的名称。

已组成 bond 且在使用中的网卡，不能用于内部通信平面，因此需要查询相应的网卡名称。

```
cd /etc/sysconfig/network
```

```
vi ifcfg-bond0
```

返回信息示例如下，可见“bond0”由“eth0”和“eth1”组成。

```
BONDING_MASTER=yes
```

```
TYPE=Bond
```

```
STARTMODE=auto
```

```
BONDING_MODULE_OPTS="mode=4 xmit_hash_policy=layer3+4 miimon=100"
```

```
NM_CONTROLLED=no
```

```
BOOTPROTO=dhcp
```

```
DEVICE=bond0
```

```
USERCONTRL=no
```

```
LLADDR=fa:16:3e:3d:1c:e0
```

```
BONDING_SLAVE1=eth1
```

```
BONDING_SLAVE0=eth0
```

查询完成后，退出。

### 步骤 6 查询所有网卡的的状态。

```
ip link
```

返回信息示例如下：

```
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN mode
DEFAULT group default
 link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
2: eth0: <BROADCAST,MULTICAST,SLAVE,UP,LOWER_UP> mtu 8888 qdisc mq master bond0
state UP mode DEFAULT group default qlen 1000
 link/ether fa:16:3e:3d:1c:e0 brd ff:ff:ff:ff:ff:ff
3: eth1: <BROADCAST,MULTICAST,SLAVE,UP,LOWER_UP> mtu 8888 qdisc mq master bond0
state UP mode DEFAULT group default qlen 1000
 link/ether fa:16:3e:3d:1c:e0 brd ff:ff:ff:ff:ff:ff
4: eth2: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state DOWN mode
```

```
DEFAULT group default qlen 1000
 link/ether 38:4c:4f:89:55:8d brd ff:ff:ff:ff:ff:ff
5: eth3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state DOWN mode
DEFAULT group default qlen 1000
 link/ether 38:4c:4f:89:55:8e brd ff:ff:ff:ff:ff:ff
6: bond0: <BROADCAST,MULTICAST,MASTER,UP,LOWER_UP> mtu 8888 qdisc noqueue state
UP mode DEFAULT group default
 link/ether fa:16:3e:3d:1c:e0 brd ff:ff:ff:ff:ff:ff
7: bond0.3133@bond0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 8888 qdisc noqueue
state UP mode DEFAULT group default
 link/ether fa:16:3e:57:87:6e brd ff:ff:ff:ff:ff:ff
```

**步骤 7** 将所有状态为“qdisc mq state DOWN”的网卡，设置为“qdisc mq state UP”，示例中为“eth2”和“eth3”。

```
ip link set eth2 up
```

```
ip link set eth3 up
```

**步骤 8** 重新查询网卡的状态。

```
ip link
```

返回信息示例如下：

```
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN mode
DEFAULT group default
 link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
2: eth0: <BROADCAST,MULTICAST,SLAVE,UP,LOWER_UP> mtu 8888 qdisc mq master bond0
state UP mode DEFAULT group default qlen 1000
 link/ether fa:16:3e:3d:1c:e0 brd ff:ff:ff:ff:ff:ff
3: eth1: <BROADCAST,MULTICAST,SLAVE,UP,LOWER_UP> mtu 8888 qdisc mq master bond0
state UP mode DEFAULT group default qlen 1000
 link/ether fa:16:3e:3d:1c:e0 brd ff:ff:ff:ff:ff:ff
4: eth2: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP mode
DEFAULT group default qlen 1000
 link/ether 38:4c:4f:89:55:8d brd ff:ff:ff:ff:ff:ff
5: eth3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP mode
DEFAULT group default qlen 1000
 link/ether 38:4c:4f:89:55:8e brd ff:ff:ff:ff:ff:ff
```

```
6: bond0: <BROADCAST,MULTICAST,MASTER,UP,LOWER_UP> mtu 8888 qdisc noqueue state
UP mode DEFAULT group default
```

```
link/ether fa:16:3e:3d:1c:e0 brd ff:ff:ff:ff:ff:ff
```

```
7: bond0.3133@bond0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 8888 qdisc noqueue
state UP mode DEFAULT group default
```

```
link/ether fa:16:3e:57:87:6e brd ff:ff:ff:ff:ff:ff
```

**步骤 1** 查看[步骤 8](#)中对应的网卡的状态，获取状态为“qdisc mq state UP”的网卡名称。

只有状态为“qdisc mq state UP”且未被使用过的网卡，才能组成 bond，示例中为“eth2”和“eth3”。

“eth2”的 LLADDR 为“38:4c:4f:89:55:8d”，“eth3”的 LLADDR 为“38:4c:4f:89:55:8e”。

**步骤 2** 创建“eth2”和“eth3”网卡的配置文件。

可通过复制已有网卡配置文件的方式快速创建。

```
cp ifcfg-eth0 ifcfg-eth2
```

```
cp ifcfg-eth1 ifcfg-eth3
```

**步骤 3** 修改“eth2”和“eth3”网卡的配置文件。

```
vi ifcfg-eth2
```

```
vi ifcfg-eth3
```

“eth2”网卡配置文件的修改示例如下。

其中，参数参数“MTU”配置为“8888”，“BOOTPROTO”需要配置为“STATIC”，参数“DEVICE”、“LLADDR”根据实际需要填写。

```
STARTMODE=auto
```

```
MTU=8888
```

```
NM_CONTROLLED=no
```

```
BOOTPROTO=STATIC
```

```
DEVICE=eth2
```

```
USERCONTRL=no
```

```
LLADDR=38:4c:4f:89:55:8d
```

```
TYPE=Ethernet
```

“eth3”网卡配置文件的修改示例如下：

```
STARTMODE=auto
```

```
MTU=8888
```

```
NM_CONTROLLED=no
BOOTPROTO=STATIC
DEVICE=eth3
USERCONTRL=no
LLADDR=38:4c:4f:89:55:8e
TYPE=Ethernet
```

修改完成后，保存并退出。

**步骤 4** 将“eth2”和“eth3”组 bond，假设为“bond1”。

创建 ifcfg-bond1 文件并修改配置。

```
cp ifcfg-bond0 ifcfg-bond1
vi ifcfg-bond1
```

“bond1”网卡配置文件的修改示例如下。

其中，参数“MTU”配置为“8888”，“BONDING\_MODULE\_OPTS”配置为“mode=1 miimon=100”，“BOOTPROTO”需要配置为“STATIC”，“DEVICE”、“BONDING\_SLAVE1”、“BONDING\_SLAVE0”、“IPADDR”、“NETMASK”、“NETWORK”根据实际需要填写，“LLADDR”配置为参数“BONDING\_SLAVE1”对应网卡的 LLADDR。

```
BONDING_MASTER=yes
TYPE=Bond
MTU=8888
STARTMODE=auto
BONDING_MODULE_OPTS="mode=1 miimon=100"
NM_CONTROLLED=no
BOOTPROTO=STATIC
DEVICE=bond1
USERCONTRL=no
LLADDR=38:4c:4f:89:55:8d
BONDING_SLAVE1=eth2
BONDING_SLAVE0=eth3
IPADDR=10.0.2.2
NETMASK=255.255.255.0
NETWORK=10.0.2.0
```

修改完成后，保存并退出。

**步骤 5** 使配置文件生效。

1. 创建临时目录，并将网络配置文件复制到该目录下。

```
mkdir /opt/tmp/
mkdir /opt/tmp/xml
cp /etc/sysconfig/network/ifcfg* /opt/tmp/
cp /etc/sysconfig/network/config /opt/tmp/
cp /etc/sysconfig/network/dhcp /opt/tmp/
```

2. 停止待组成 bond1 的网卡。

```
ip link set eth2 down
ip link set eth3 down
```

3. 将网卡配置文件转换成操作系统可辨识的配置文件。

```
/usr/sbin/wicked --log-target=stderr --log-level=debug3 --debug all convert --
output /opt/tmp/xml /opt/tmp/
```

4. 重新启用待组成 bond1 的网卡。

```
ip link set eth2 up
/usr/sbin/wicked --log-target=stderr --log-level=debug3 --debug all ifup --
ifconfig /opt/tmp/xml/eth2.xml eth2
ip link set eth3 up
/usr/sbin/wicked --log-target=stderr --log-level=debug3 --debug all ifup --
ifconfig /opt/tmp/xml/eth3.xml eth3
/usr/sbin/wicked --log-target=stderr --log-level=debug3 --debug all ifup --
ifconfig /opt/tmp/xml/bond1.xml bond1
```

- 步骤 6** 重新查询 IP 地址信息，可查看到 IP 地址已分配。

```
ip addr show
```

示例如下：

```
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group
default
 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,SLAVE,UP,LOWER_UP> mtu 8888 qdisc mq master bond0
state UP group default qlen 1000
 link/ether fa:16:3e:3d:1c:e0 brd ff:ff:ff:ff:ff:ff
```

```
3: eth1: <BROADCAST,MULTICAST,SLAVE,UP,LOWER_UP> mtu 8888 qdisc mq master bond0
state UP group default qlen 1000
 link/ether fa:16:3e:3d:1c:e0 brd ff:ff:ff:ff:ff:ff
4: eth2: <BROADCAST,MULTICAST,SLAVE,UP,LOWER_UP> mtu 8888 qdisc mq master bond1
state UP group default qlen 1000
 link/ether 38:4c:4f:89:55:8d brd ff:ff:ff:ff:ff:ff
5: eth3: <BROADCAST,MULTICAST,SLAVE,UP,LOWER_UP> mtu 8888 qdisc mq master bond1
state UP group default qlen 1000
 link/ether 38:4c:4f:89:55:8d brd ff:ff:ff:ff:ff:ff
6: bond0: <BROADCAST,MULTICAST,MASTER,UP,LOWER_UP> mtu 8888 qdisc noqueue state
UP group default
 link/ether fa:16:3e:3d:1c:e0 brd ff:ff:ff:ff:ff:ff
 inet 10.0.1.2/24 brd 10.0.1.255 scope global bond0
 valid_lft forever preferred_lft forever
 inet6 fe80::f816:3eff:fe3d:1ce0/64 scope link
 valid_lft forever preferred_lft forever
7: bond0.3133@bond0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 8888 qdisc noqueue
state UP group default
 link/ether fa:16:3e:57:87:6e brd ff:ff:ff:ff:ff:ff
 inet 10.0.3.2/24 brd 10.0.2.255 scope global bond0.3133
 valid_lft forever preferred_lft forever
 inet6 fe80::f816:3eff:fe57:876e/64 scope link
 valid_lft forever preferred_lft forever
8: bond1: <BROADCAST,MULTICAST,MASTER,UP,LOWER_UP> mtu 8888 qdisc noqueue state
UP group default
 link/ether 38:4c:4f:89:55:8d brd ff:ff:ff:ff:ff:ff
 inet 10.0.2.2/24 brd 10.0.2.255 scope global bond1
 valid_lft forever preferred_lft forever
 inet6 fe80::3a4c:4fff:fe29:b36/64 scope link
 valid_lft forever preferred_lft forever
```

**步骤 7** 删除创建的临时目录。

```
cd /opt
rm -rf tmp/
```

**步骤 8** 参考上述步骤，完成其他物理机上的配置。

——结束

## 4.12.2 SUSE Linux Enterprise Server 11 系列

本章节主要介绍 SUSE Linux Enterprise Server 11 系列自定义 VLAN 网络配置。

### 注意事项

自定义 VLAN 网络网段不能与现有的物理机上已经配置的网段重叠。

下面以 CentOS 6.8 (x86\_64) 操作系统为例，举例介绍物理机的自定义 VLAN 网络配置方法：

### 说明

RedHat 系列、Oracle Linux 系列、Euler 系列及 CentOS 系列操作系统的配置方法类似。

**步骤 1** 以“root”用户，使用密钥或密码登录物理机。

**步骤 2** 进入物理机的命令行界面，查询网卡信息。

```
ip link
```

返回信息示例如下：

```
bms-multinics-test-0002:/etc/sysconfig/network # ip link
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 16436 qdisc noqueue state UNKNOWN
 link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
2: eth0: <BROADCAST,MULTICAST,SLAVE,UP,LOWER_UP> mtu 8888 qdisc mq master bond0 state UP qlen 1000
 link/ether fa:16:3e:0d:13:7c brd ff:ff:ff:ff:ff:ff
3: eth1: <BROADCAST,MULTICAST,SLAVE,UP,LOWER_UP> mtu 8888 qdisc mq master bond0 state UP qlen 1000
 link/ether fa:16:3e:0d:13:7c brd ff:ff:ff:ff:ff:ff
4: eth4: <BROADCAST,MULTICAST,SLAVE,UP,LOWER_UP> mtu 8888 qdisc mq master bond1 state UP qlen 1000
 link/ether 40:7d:0f:f4:ff:5c brd ff:ff:ff:ff:ff:ff
5: eth5: <BROADCAST,MULTICAST,SLAVE,UP,LOWER_UP> mtu 8888 qdisc mq master bond1 state UP qlen 1000
 link/ether 40:7d:0f:f4:ff:5c brd ff:ff:ff:ff:ff:ff
6: bond0: <BROADCAST,MULTICAST,MASTER,UP,LOWER_UP> mtu 8888 qdisc noqueue state UP
 link/ether fa:16:3e:0d:13:7c brd ff:ff:ff:ff:ff:ff
7: bond1: <BROADCAST,MULTICAST,MASTER,UP,LOWER_UP> mtu 8888 qdisc noqueue state UP
 link/ether 40:7d:0f:f4:ff:5c brd ff:ff:ff:ff:ff:ff
```

## 说明

其中，“eth0”和“eth1”为承载 VPC 网络的网络设备，“eth3”和“eth5”为承载自定义 VLAN 网络的网络设备。

**步骤 3** 执行以下命令，查看“/etc/udev/rules.d/”目录下是否有“80-persistent-net.rules”配置文件。

```
ll /etc/udev/rules.d/ | grep 80-persistent-net.rules
```

- 如果存在“80-persistent-net.rules”，且该配置文件中已存在[步骤 2](#)中查询到的除“bond0”和“lo”以外的其它所有网卡和对应的 MAC 地址，请执行[步骤 6](#)。
- 否则，继续执行[步骤 4](#)。

**步骤 4** 执行以下命令，将“/etc/udev/rules.d/70-persistent-net.rules”文件拷贝一份（文件名为“/etc/udev/rules.d/80-persistent-net.rules”）。

```
cp -p /etc/udev/rules.d/70-persistent-net.rules /etc/udev/rules.d/80-persistent-net.rules
```

**步骤 5** 设置 udev 规则。

将[步骤 2](#)中查询到的除“eth0”和“eth1”以外的网卡（即“/etc/udev/rules.d/70-persistent-net.rules”中未体现的网卡 MAC 地址和名称），写入

“/etc/udev/rules.d/80-persistent-net.rules”文件中，使得物理机重启复位后，网卡名称和顺序不会发生改变。

## 说明

网卡的 MAC 地址和名称中的字母，请使用小写字母。

```
vim /etc/udev/rules.d/80-persistent-net.rules
```

修改后的示例如下：

```
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="?*", ATTR{address}=="e8:4d:d0:c8:99:67", NAME="eth0"
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="?*", ATTR{address}=="e8:4d:d0:c8:99:68", NAME="eth1"
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="?*", ATTR{address}=="40:7d:0f:f4:ff:5c", NAME="eth3"
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="?*", ATTR{address}=="40:7d:0f:f4:ff:5d", NAME="eth5"
```

修改完成后，按“Esc”，输入:wq 保存并退出。

**步骤 6** 执行以下命令，将网络配置文件“/etc/sysconfig/network-scripts/ifcfg-bond0”拷贝为“/etc/sysconfig/network-scripts/ifcfg-bond1”，将网络配置文件“/etc/sysconfig/network-scripts/ifcfg-eth0”拷贝为“/etc/sysconfig/network-scripts/ifcfg-eth3”和“/etc/sysconfig/network-scripts/ifcfg-eth5”。

```
cp -p /etc/sysconfig/network-scripts/ifcfg-bond0 /etc/sysconfig/network-scripts/ifcfg-bond1
```

```
cp -p /etc/sysconfig/network-scripts/ifcfg-eth0 /etc/sysconfig/network-scripts/ifcfg-eth3
```

```
cp -p /etc/sysconfig/network-scripts/ifcfg-eth0 /etc/sysconfig/network-scripts/ifcfg-eth5
```

**步骤 7** 执行以下命令，编辑“/etc/sysconfig/network-scripts/ifcfg-eth3”和“/etc/sysconfig/network-scripts/ifcfg-eth5”，配置“eth3”设备和“eth5”设备的网络配置文件。

```
vim /etc/sysconfig/network-scripts/ifcfg-eth3
```

“eth3”按以下格式编辑：

```
USERCTL=no
MTU=8888
NM_CONTROLLED=no
BOOTPROTO=static
DEVICE=eth3
TYPE=Ethernet
ONBOOT=yes
MASTER=bond1
SLAVE=yes
```

其中，“BOOTPROTO”参数取值修改为“static”，“DEVICE”为对应的网络设备名称，取值即为“eth3”，“MASTER”为对应的自定义 VLAN 网络端口的名称，取值如“bond1”，其他参数可保持不变。

```
vim /etc/sysconfig/network-scripts/ifcfg-eth5
```

“eth5”按以下格式编辑（格式和规则和“eth3”一致）：

```
USERCTL=no
MTU=8888
NM_CONTROLLED=no
BOOTPROTO=static
DEVICE=eth5
TYPE=Ethernet
ONBOOT=yes
MASTER=bond1
SLAVE=yes
```

**步骤 8** 执行以下命令，编辑“/etc/sysconfig/network-scripts/ifcfg-bond1”。

```
vim /etc/sysconfig/network-scripts/ifcfg-bond1
```

按以下格式编辑：

```
MACADDR=f4:4c:7f:3f:da:07
BONDING_MASTER=yes
USERCTL=no
ONBOOT=yes
NM_CONTROLLED=no
BOOTPROTO=static
BONDING_OPTS="mode=1 miimon=100"
DEVICE=bond1
TYPE=Bond
IPADDR=10.10.10.3
NETMASK=255.255.255.0
MTU=8888
```

其中，

- “MACADDR” 参数取值修改为自定义 VLAN 网络 “eth3” 或者 “eth5” 设备的 MAC 地址。
- “BOOTPROTO” 参数取值修改为 “static”。
- “DEVICE” 参数取值修改为 “bond1”。
- “IPADDR” 参数取值修改为待给 “bond1” 分配的 IP 地址（为自定义 VLAN 网络规划的 IP 地址在没有与 VPC 网段冲突的情况下可任意规划，需要通过自定义 VLAN 网络通信的物理机须将自定义 VLAN 网络配置在同一个网段），如 “10.10.10.3”。
- “NETMASK” 参数为给自定义 VLAN 网络 “bond1” 配置的 IP 的子网掩码。

其他参数可保持不变。

修改完成后，按 “Esc”，输入 `:wq` 保存并退出。

**步骤 9** 执行以下命令，重启网络。

```
ifup eth4
```

```
ifup eth5
```

```
ifup bond1
```

```
bms-multinics-test-0002:/etc/sysconfig/network # ifup eth4
eth4 device: Intel Corporation 82599ES 10-Gigabit SFI/SFP+ Network Connection (rev 01)
bms-multinics-test-0002:/etc/sysconfig/network # ifup eth5
eth5 device: Intel Corporation 82599ES 10-Gigabit SFI/SFP+ Network Connection (rev 01)
bms-multinics-test-0002:/etc/sysconfig/network # ifup bond1
bond1
bond1 enslaved interface: eth5
bond1 enslaved interface: eth4
bms-multinics-test-0002:/etc/sysconfig/network # █
```

## 说明

其中，“eth4”和“eth5”分别为承载自定义 VLAN 网络的网口，“bond1”为自定义 VLAN 网络对应的端口组。

**步骤 10** 执行以下命令，查看网卡设备的状态和“bond1”配置文件是否生效。

```
ifup bond1
```

```
ip link
```

```
bms-multinics-test-0002:/etc/sysconfig/network # ip link
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 16436 qdisc noqueue state UNKNOWN
 link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
2: eth0: <BROADCAST,MULTICAST,SLAVE,UP,LOWER_UP> mtu 8888 qdisc mq master bond0 state UP qlen 1000
 link/ether fa:16:3e:0d:13:7c brd ff:ff:ff:ff:ff:ff
3: eth1: <BROADCAST,MULTICAST,SLAVE,UP,LOWER_UP> mtu 8888 qdisc mq master bond0 state UP qlen 1000
 link/ether fa:16:3e:0d:13:7c brd ff:ff:ff:ff:ff:ff
4: eth4: <BROADCAST,MULTICAST,SLAVE,UP,LOWER_UP> mtu 8888 qdisc mq master bond1 state UP qlen 1000
 link/ether 40:7d:0f:f4:ff:5c brd ff:ff:ff:ff:ff:ff
5: eth5: <BROADCAST,MULTICAST,SLAVE,UP,LOWER_UP> mtu 8888 qdisc mq master bond1 state UP qlen 1000
 link/ether 40:7d:0f:f4:ff:5c brd ff:ff:ff:ff:ff:ff
6: bond0: <BROADCAST,MULTICAST,MASTER,UP,LOWER_UP> mtu 8888 qdisc noqueue state UP
 link/ether fa:16:3e:0d:13:7c brd ff:ff:ff:ff:ff:ff
7: bond1: <BROADCAST,MULTICAST,MASTER,UP,LOWER_UP> mtu 8888 qdisc noqueue state UP
 link/ether 40:7d:0f:f4:ff:5c brd ff:ff:ff:ff:ff:ff
```

```
ifconfig
```

```

bms-multinics-test-0002:/etc/sysconfig/network # ifconfig
bond0 Link encap:Ethernet HWaddr FA:16:3E:0D:13:7C
 inet addr:192.168.20.143 Bcast:192.168.20.255 Mask:255.255.255.0
 inet6 addr: fe80::f816:3eff:fe0d:137c/64 Scope:Link
 UP BROADCAST RUNNING MASTER MULTICAST MTU:8888 Metric:1
 RX packets:5300 errors:0 dropped:1627 overruns:0 frame:0
 TX packets:1926 errors:0 dropped:0 overruns:0 carrier:0
 collisions:0 txqueuelen:0
 RX bytes:392043 (382.8 Kb) TX bytes:424419 (414.4 Kb)

bond1 Link encap:Ethernet HWaddr 40:7D:0F:F4:FF:5C
 inet addr:10.10.10.4 Bcast:10.10.10.255 Mask:255.255.255.0
 inet6 addr: fe80::427d:fff:fef4:ff5c/64 Scope:Link
 UP BROADCAST RUNNING MASTER MULTICAST MTU:8888 Metric:1
 RX packets:0 errors:0 dropped:0 overruns:0 frame:0
 TX packets:15 errors:0 dropped:0 overruns:0 carrier:0
 collisions:0 txqueuelen:0
 RX bytes:0 (0.0 b) TX bytes:1194 (1.1 Kb)

eth0 Link encap:Ethernet HWaddr FA:16:3E:0D:13:7C
 UP BROADCAST RUNNING SLAVE MULTICAST MTU:8888 Metric:1
 RX packets:3673 errors:0 dropped:0 overruns:0 frame:0
 TX packets:1926 errors:0 dropped:0 overruns:0 carrier:0
 collisions:0 txqueuelen:1000
 RX bytes:293157 (286.2 Kb) TX bytes:424419 (414.4 Kb)

eth1 Link encap:Ethernet HWaddr FA:16:3E:0D:13:7C
 UP BROADCAST RUNNING SLAVE MULTICAST MTU:8888 Metric:1
 RX packets:1627 errors:0 dropped:1627 overruns:0 frame:0
 TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
 collisions:0 txqueuelen:1000
 RX bytes:98886 (96.5 Kb) TX bytes:0 (0.0 b)

eth4 Link encap:Ethernet HWaddr 40:7D:0F:F4:FF:5C
 UP BROADCAST RUNNING SLAVE MULTICAST MTU:8888 Metric:1
 RX packets:0 errors:0 dropped:0 overruns:0 frame:0
 TX packets:11 errors:0 dropped:0 overruns:0 carrier:0
 collisions:0 txqueuelen:1000
 RX bytes:0 (0.0 b) TX bytes:866 (866.0 b)

eth5 Link encap:Ethernet HWaddr 40:7D:0F:F4:FF:5C
 UP BROADCAST RUNNING SLAVE MULTICAST MTU:8888 Metric:1
 RX packets:0 errors:0 dropped:0 overruns:0 frame:0
 TX packets:4 errors:0 dropped:0 overruns:0 carrier:0
 collisions:0 txqueuelen:1000
 RX bytes:0 (0.0 b) TX bytes:328 (328.0 b)

lo Link encap:Local Loopback
 inet addr:127.0.0.1 Mask:255.0.0.0

```

**步骤 11** 参见上述步骤，完成其他物理机的配置。

**步骤 12** 待其他物理机配置完成后，互相 ping 对端自定义 VLAN 网络配置的同网段 IP，检查是否可以 ping 通。

```

[root@bms-qinq-demo ~]# tcpdump -i bond1 -mne host 10.10.10.4
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on bond1, link-type EN10MB (Ethernet), capture size 96 bytes
18:51:55.196928 40:7d:0f:f4:ff:5c > ff:ff:ff:ff:ff:ff, ethertype ARP (0x0806), length 60: arp who-has 10.10.10.3 tel
l 10.10.10.4
18:51:55.196951 f4:4c:7f:3f:da:07 > 40:7d:0f:f4:ff:5c, ethertype ARP (0x0806), length 42: arp reply 10.10.10.3 is-at
f4:4c:7f:3f:da:07
18:51:55.197005 40:7d:0f:f4:ff:5c > f4:4c:7f:3f:da:07, ethertype IPv4 (0x0800), length 98: 10.10.10.4 > 10.10.10.3:
ICMP echo request, id 25888, seq 1, length 64
18:51:55.197031 f4:4c:7f:3f:da:07 > 40:7d:0f:f4:ff:5c, ethertype IPv4 (0x0800), length 98: 10.10.10.3 > 10.10.10.4:
ICMP echo reply, id 25888, seq 1, length 64
18:51:56.196847 40:7d:0f:f4:ff:5c > f4:4c:7f:3f:da:07, ethertype IPv4 (0x0800), length 98: 10.10.10.4 > 10.10.10.3:
ICMP echo request, id 25889, seq 2, length 64
18:51:56.196852 f4:4c:7f:3f:da:07 > 40:7d:0f:f4:ff:5c, ethertype IPv4 (0x0800), length 98: 10.10.10.3 > 10.10.10.4:
PING 10.10.10.3 (10.10.10.3) 56(84) bytes of data.
64 bytes from 10.10.10.3: icmp_seq=1 ttl=64 time=0.546 ms
64 bytes from 10.10.10.3: icmp_seq=2 ttl=64 time=0.047 ms
64 bytes from 10.10.10.3: icmp_seq=3 ttl=64 time=0.040 ms
64 bytes from 10.10.10.3: icmp_seq=4 ttl=64 time=0.038 ms
64 bytes from 10.10.10.3: icmp_seq=5 ttl=64 time=0.036 ms
64 bytes from 10.10.10.3: icmp_seq=6 ttl=64 time=0.035 ms
64 bytes from 10.10.10.3: icmp_seq=7 ttl=64 time=0.038 ms
64 bytes from 10.10.10.3: icmp_seq=8 ttl=64 time=0.036 ms
^C
--- 10.10.10.3 ping statistics ---
8 packets transmitted, 8 received, 0% packet loss, time 7000ms
rtt min/avg/max/mdev = 0.035/0.102/0.546/0.167 ms

```

### 4.12.3 RedHat 系列，CentOS 系列，Oracle Linux 系列，Euler 系列

本章节主要介绍 RedHat 系列，CentOS 系列，Oracle Linux 系列，Euler 系列自定义 VLAN 网络配置。

#### 注意事项

自定义 VLAN 网络网段不能与现有的物理机上已经配置的网段重叠。

下面以 CentOS 6.8 (x86\_64) 操作系统为例，举例介绍物理机的自定义 VLAN 网络配置方法：

#### 说明

RedHat 系列、Oracle Linux 系列、Euler 系列及 CentOS 系列操作系统的配置方法类似。

**步骤 1** 以“root”用户，使用密钥或密码登录物理机。

**步骤 2** 进入物理机的命令行界面，查询网卡信息。

ip link

返回信息示例如下：

```

[root@bms-qinq-demo ~]# ip link
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN
 link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
2: eth0: <BROADCAST,MULTICAST,SLAVE,UP,LOWER_UP> mtu 8888 qdisc mq master bond0 state UP qlen 1000
 link/ether fa:16:3e:e5:ec:6a brd ff:ff:ff:ff:ff:ff
3: eth1: <BROADCAST,MULTICAST,SLAVE,UP,LOWER_UP> mtu 8888 qdisc mq master bond0 state UP qlen 1000
 link/ether fa:16:3e:e5:ec:6a brd ff:ff:ff:ff:ff:ff
4: eth3: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN qlen 1000
 link/ether f4:4c:7f:3f:da:07 brd ff:ff:ff:ff:ff:ff
5: eth5: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN qlen 1000
 link/ether f4:4c:7f:3f:da:08 brd ff:ff:ff:ff:ff:ff
6: bond0: <BROADCAST,MULTICAST,PROMISC,MASTER,UP,LOWER_UP> mtu 8888 qdisc noqueue state UP
 link/ether fa:16:3e:e5:ec:6a brd ff:ff:ff:ff:ff:ff
[root@bms-qinq-demo ~]#

```

## 说明

其中，“eth0”和“eth1”为承载 VPC 网络的网络设备，“eth3”和“eth5”为承载自定义 VLAN 网络的网络设备。

**步骤 3** 执行以下命令，查看“/etc/udev/rules.d/”目录下是否有“80-persistent-net.rules”配置文件。

```
ll /etc/udev/rules.d/ | grep 80-persistent-net.rules
```

1 如果存在“80-persistent-net.rules”，且该配置文件中已存在[步骤 2](#)中查询到的除“bond0”和“lo”以外的其它所有网卡和对应的 MAC 地址，请执行[步骤 6](#)。

1 否则，继续执行[步骤 4](#)。

**步骤 4** 执行以下命令，将“/etc/udev/rules.d/70-persistent-net.rules”文件拷贝一份（文件名为“/etc/udev/rules.d/80-persistent-net.rules”）。

```
cp -p /etc/udev/rules.d/70-persistent-net.rules /etc/udev/rules.d/80-persistent-net.rules
```

**步骤 5** 设置 udev 规则。

将[步骤 2](#)中查询到的除“eth0”和“eth1”以外的网卡（即“/etc/udev/rules.d/70-persistent-net.rules”中未体现的网卡 MAC 地址和名称），写入“/etc/udev/rules.d/80-persistent-net.rules”文件中，使得物理机重启复位后，网卡名称和顺序不会发生改变。

## 说明

网卡的 MAC 地址和名称中的字母，请使用小写字母。

```
vim /etc/udev/rules.d/80-persistent-net.rules
```

修改后的示例如下：

```
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="*", ATTR{address}=="e8:4d:d0:c8:99:5b", NAME="eth0"
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="*", ATTR{address}=="e8:4d:d0:c8:99:5c", NAME="eth1"
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="*", ATTR{address}=="f4:4c:7f:3f:da:07", NAME="eth3"
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="*", ATTR{address}=="f4:4c:7f:3f:da:08", NAME="eth5"
```

修改完成后，按“Esc”，输入:wq 保存并退出。

**步骤 6** 执行以下命令，将网络配置文件“/etc/sysconfig/network-scripts/ifcfg-bond0”拷贝为“/etc/sysconfig/network-scripts/ifcfg-bond1”，将网络配置文件“/etc/sysconfig/network-scripts/ifcfg-eth0”拷贝为“/etc/sysconfig/network-scripts/ifcfg-eth3”和“/etc/sysconfig/network-scripts/ifcfg-eth5”。

```
cp -p /etc/sysconfig/network-scripts/ifcfg-bond0 /etc/sysconfig/network-scripts/ifcfg-bond1
```

```
cp -p /etc/sysconfig/network-scripts/ifcfg-eth0 /etc/sysconfig/network-scripts/ifcfg-eth3
```

```
cp -p /etc/sysconfig/network-scripts/ifcfg-eth0 /etc/sysconfig/network-scripts/ifcfg-eth5
```

**步骤 7** 执行以下命令，编辑“/etc/sysconfig/network-scripts/ifcfg-eth3”和“/etc/sysconfig/network-scripts/ifcfg-eth5”，配置“eth3”设备和“eth5”设备的网络配置文件。

```
vim /etc/sysconfig/network-scripts/ifcfg-eth3
```

“eth3”按以下格式编辑：

```
USERCTL=no
MTU=8888
NM_CONTROLLED=no
BOOTPROTO=static
DEVICE=eth3
TYPE=Ethernet
ONBOOT=yes
```

```
MASTER=bond1
SLAVE=yes
```

其中，“BOOTPROTO”参数取值修改为“static”，“DEVICE”为对应的网络设备名称，取值即为“eth3”，“MASTER”为对应的自定义 VLAN 网络端口的名称，取值如“bond1”，其他参数可保持不变。

```
vim /etc/sysconfig/network-scripts/ifcfg-eth5
```

“eth5”按以下格式编辑（格式和规则和“eth3”一致）：

```
USERCTL=no
MTU=8888
NM_CONTROLLED=no
BOOTPROTO=static
DEVICE=eth5
TYPE=Ethernet
ONBOOT=yes
MASTER=bond1
SLAVE=yes
```

**步骤 8** 执行以下命令，编辑“/etc/sysconfig/network-scripts/ifcfg-bond1”。

```
vim /etc/sysconfig/network-scripts/ifcfg-bond1
```

按以下格式编辑：

```
MACADDR=f4:4c:7f:3f:da:07
BONDING_MASTER=yes
USERCTL=no
ONBOOT=yes
NM_CONTROLLED=no
BOOTPROTO=static
BONDING_OPTS="mode=1 miimon=100"
DEVICE=bond1
TYPE=Bond
```

```
IPADDR=10.10.10.3
NETMASK=255.255.255.0
MTU=8888
```

其中，

- “MACADDR” 参数取值修改为自定义 VLAN 网络 “eth3” 或者 “eth5” 设备的 MAC 地址。
- “BOOTPROTO” 参数取值修改为 “static”。
- “DEVICE” 参数取值修改为 “bond1”。
- “IPADDR” 参数取值修改为待给 “bond1” 分配的 IP 地址（为自定义 VLAN 网络规划的 IP 地址在没有与 VPC 网段冲突的情况下可任意规划，需要通过自定义 VLAN 网络通信的物理机须将自定义 VLAN 网络配置在同一个网段），如 “10.10.10.3”。
- “NETMASK” 参数为给自定义 VLAN 网络 “bond1” 配置的 IP 的子网掩码。

其他参数可保持不变。

修改完成后，按 “Esc”，输入 :wq 保存并退出。

**步骤 9** 执行以下命令，启动自定义 VLAN 网络端口组 “bond1”。

```
ifup bond1
```

```
[root@bms-qinq-demo network-scripts]# ifup bond1
Determining if ip address 10.10.10.3 is already in use for device bond1...
[root@bms-qinq-demo network-scripts]# ip link
```

**步骤 10** 参见上述步骤，完成其他物理机的配置。

**步骤 11** 待其他物理机配置完成后，互相 ping 对端自定义 VLAN 网络配置的同网段 IP，检查是否可以 ping 通。



The image shows two terminal windows. The left window displays the output of the command `tcpdump -i bond1 -nne host 10.10.10.4`, showing ARP and ICMP traffic between the two hosts. The right window displays the output of the command `ping 10.10.10.3`, showing successful ping results with 8 packets transmitted and 0% packet loss.

——结束

## 4.12.4 Ubuntu 系列

本章节主要介绍 Ubuntu 系列自定义 VLAN 网络配置。

### 注意事项

自定义 VLAN 网络网段不能与现有的物理机上已经配置的网段重叠。

下面以 Ubuntu 16.04 LTS (Xenial Xerus x86\_64) 操作系统为例，举例介绍物理机的自定义 VLAN 网络配置方法：

### 说明

Ubuntu 系列其他操作系统的配置方法与 Ubuntu 16.04 LTS (Xenial Xerus x86\_64) 类似。

**步骤 1** 以“root”用户，使用密钥或密码登录物理机。

**步骤 2** 进入物理机的命令行界面，查询网卡信息。

`ip link`

返回信息示例如下：

```
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN mode
DEFAULT group default qlen 1
 link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
2: eth0: <BROADCAST,MULTICAST,SLAVE,UP,LOWER_UP> mtu 8888 qdisc mq master
bond0 state UP mode DEFAULT group default qlen 1000
 link/ether fa:16:3e:1c:35:37 brd ff:ff:ff:ff:ff:ff
3: eth1: <BROADCAST,MULTICAST,SLAVE,UP,LOWER_UP> mtu 8888 qdisc mq master
bond0 state UP mode DEFAULT group default qlen 1000
 link/ether fa:16:3e:1c:35:37 brd ff:ff:ff:ff:ff:ff
4: enp129s0f0: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN mode
```

```
DEFAULT group default qlen 1000
 link/ether f4:4c:7f:3f:da:07 brd ff:ff:ff:ff:ff:ff
5: enp129s0f1: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN mode
DEFAULT group default qlen 1000
 link/ether f4:4c:7f:3f:da:08 brd ff:ff:ff:ff:ff:ff
6: bond0: <BROADCAST,MULTICAST,MASTER,UP,LOWER_UP> mtu 8888 qdisc noqueue
state UP mode DEFAULT group default qlen 1000
 link/ether fa:16:3e:1c:35:37 brd ff:ff:ff:ff:ff:ff
```

## 说明

其中，“eth0”和“eth1”为承载VPC网络的网络设备，“enp129s0f0”和“enp129s0f1”为承载自定义VLAN网络的网络设备。下面步骤将使用“enp129s0f0”和“enp129s0f1”配置自定义VLAN网络。

**步骤 3** 执行以下命令，查看“/etc/udev/rules.d/”目录下是否有“80-persistent-net.rules”配置文件。

```
ll /etc/udev/rules.d/ | grep 80-persistent-net.rules
```

1 如果存在“80-persistent-net.rules”，且该配置文件中已存在[步骤 2](#)中查询到的除“bond0”和“lo”以外的其它所有网卡和对应的MAC地址，请执行[步骤 6](#)。

1 否则，继续执行[步骤 4](#)。

**步骤 4** 执行以下命令，将“/etc/udev/rules.d/70-persistent-net.rules”文件拷贝一份（文件名为“/etc/udev/rules.d/80-persistent-net.rules”）。

```
cp -p /etc/udev/rules.d/70-persistent-net.rules /etc/udev/rules.d/80-persistent-net.rules
```

**步骤 5** 设置udev规则。

将[步骤 2](#)中查询到的除“lo”、“eth0”、“eth1”和“bond0”以外的网卡和 MAC 对应关系添加到“/etc/udev/rules.d/80-persistent-net.rules”文件中，使得物理机重启复位后，网卡名称和顺序不会发生改变。

## 说明

网卡的 MAC 地址和名称中的字母，请使用小写字母。

```
vim /etc/udev/rules.d/80-persistent-net.rules
```

修改后的示例如下：

```
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="?*",
ATTR{address}=="e8:4d:d0:c8:99:5b", NAME="eth0"
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="?*",
ATTR{address}=="e8:4d:d0:c8:99:5c", NAME="eth1"
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="?*",
ATTR{address}=="f4:4c:7f:3f:da:07", NAME="enp129s0f0"
SUBSYSTEM=="net", ACTION=="add", DRIVERS=="?*",
ATTR{address}=="f4:4c:7f:3f:da:08", NAME="enp129s0f1"
```

修改完成后，按“Esc”，输入:wq 保存并退出。

**步骤 6** 执行以下命令，将网卡配置文件“/etc/network/interfaces.d/50-cloud-init.cfg”拷贝为“/etc/network/interfaces.d/60-cloud-init.cfg”。

```
cp -p /etc/network/interfaces.d/50-cloud-init.cfg
/etc/network/interfaces.d/60-cloud-init.cfg
```

**步骤 7** 执行以下命令，编辑“/etc/network/interfaces.d/60-cloud-init.cfg”，配置“enp129s0f0”设备和“enp129s0f1”设备的网络配置文件“/etc/network/interfaces.d/60-cloud-init.cfg”。

```
vim /etc/network/interfaces.d/60-cloud-init.cfg
```

按如下格式编辑：

```
auto enp129s0f0
iface enp129s0f0 inet manual
bond_mode 1
bond-master bond1
bond_miimon 100
mtu 8888
auto enp129s0f1
iface enp129s0f1 inet manual
bond_mode 1
bond-master bond1
bond_miimon 100
mtu 8888
auto bond1
iface bond1 inet static
bond_miimon 100
bond-slaves none
bond_mode 1
address 10.10.10.3
netmask 255.255.255.0
hwaddress f4:4c:7f:3f:da:07
mtu 8888
```

其中，

- “enp129s0f0”和“enp129s0f1”为承载自定义 VLAN 网络配置的网卡名称。
- “hwaddress”为“enp129s0f0”设备对应的 MAC 地址。
- “address”的取值为给自定义 VLAN 网络“bond1”配置的 IP（给自定义 VLAN 网络规划的 IP 地址在没有与 VPC 网段冲突的情况下可任意规划，需要通过自定义 VLAN 网络通信的物理机须将自定义 VLAN 网络配置在同一个网段）。
- “netmask”的取值为给自定义 VLAN 网络“bond1”配置的 IP 的掩码。
- 各个设备的其他参数可参考如上信息进行配置，如“mtu”配置为“8888”，“bond\_miimon”配置为“100”，“bond\_mode”配置为“1”等。

修改完成后，按“Esc”，输入:wq 保存并退出。

**步骤 8** 执行以下命令，重启网络。

```
ifup enp129s0f0
```

```
ifup enp129s0f1
```

## 说明

其中，“enp129s0f0”和“enp129s0f1”分别为承载自定义 VLAN 网络的网卡。

**步骤 9** 执行以下命令，查看网卡设备的状态和“bond1”配置文件是否生效。

```
ip link
```

```
root@bms-af1d:~# ip link
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN mode DEFAULT group default qlen 1
 link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
2: eth0: <BROADCAST,MULTICAST,SLAVE,UP,LOWER_UP> mtu 8888 qdisc mq master bond0 state UP mode DEFAULT group default qlen 1000
 link/ether fa:16:3e:1c:35:37 brd ff:ff:ff:ff:ff:ff
3: eth1: <BROADCAST,MULTICAST,SLAVE,UP,LOWER_UP> mtu 8888 qdisc mq master bond0 state UP mode DEFAULT group default qlen 1000
 link/ether fa:16:3e:1c:35:37 brd ff:ff:ff:ff:ff:ff
4: enp129s0f0: <BROADCAST,MULTICAST,SLAVE,UP,LOWER_UP> mtu 8888 qdisc mq master bond1 state UP mode DEFAULT group default qlen 1000
 link/ether f4:4c:7f:3f:da:07 brd ff:ff:ff:ff:ff:ff
5: enp129s0f1: <BROADCAST,MULTICAST,SLAVE,UP,LOWER_UP> mtu 8888 qdisc mq master bond1 state UP mode DEFAULT group default qlen 1000
 link/ether f4:4c:7f:3f:da:07 brd ff:ff:ff:ff:ff:ff
7: bond0: <BROADCAST,MULTICAST,MASTER,UP,LOWER_UP> mtu 8888 qdisc noqueue state UP mode DEFAULT group default qlen 1000
 link/ether fa:16:3e:1c:35:37 brd ff:ff:ff:ff:ff:ff
8: bond1: <BROADCAST,MULTICAST,MASTER,UP,LOWER_UP> mtu 8888 qdisc noqueue state UP mode DEFAULT group default qlen 1000
 link/ether f4:4c:7f:3f:da:07 brd ff:ff:ff:ff:ff:ff
root@bms-af1d:~#
```

```
ifconfig
```

```
root@bms-afld:~# ifconfig
bond0 Link encap:Ethernet HWaddr fa:16:3e:1c:35:37
 inet addr:192.168.20.195 Bcast:192.168.20.255 Mask:255.255.255.0
 inet6 addr: fe80::f816:3eff:fe1c:3537/64 Scope:Link
 UP BROADCAST RUNNING MASTER MULTICAST MTU:8888 Metric:1
 RX packets:77 errors:0 dropped:18 overruns:0 frame:0
 TX packets:74 errors:0 dropped:0 overruns:0 carrier:0
 collisions:0 txqueuelen:1000
 RX bytes:6569 (6.5 KB) TX bytes:12236 (12.2 KB)

bond1 Link encap:Ethernet HWaddr f4:4c:7f:3f:da:07
 inet addr:10.10.10.3 Bcast:10.10.10.255 Mask:255.255.255.0
 inet6 addr: fe80::f64c:7fff:fe3f:da07/64 Scope:Link
 UP BROADCAST RUNNING MASTER MULTICAST MTU:8888 Metric:1
 RX packets:0 errors:0 dropped:0 overruns:0 frame:0
 TX packets:10 errors:0 dropped:0 overruns:0 carrier:0
 collisions:0 txqueuelen:1000
 RX bytes:0 (0.0 B) TX bytes:776 (776.0 B)

enp129s0f0 Link encap:Ethernet HWaddr f4:4c:7f:3f:da:07
 UP BROADCAST RUNNING SLAVE MULTICAST MTU:8888 Metric:1
 RX packets:0 errors:0 dropped:0 overruns:0 frame:0
 TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
 collisions:0 txqueuelen:1000
 RX bytes:0 (0.0 B) TX bytes:0 (0.0 B)

enp129s0f1 Link encap:Ethernet HWaddr f4:4c:7f:3f:da:07
 UP BROADCAST RUNNING SLAVE MULTICAST MTU:8888 Metric:1
 RX packets:0 errors:0 dropped:0 overruns:0 frame:0
 TX packets:10 errors:0 dropped:0 overruns:0 carrier:0
 collisions:0 txqueuelen:1000
 RX bytes:0 (0.0 B) TX bytes:776 (776.0 B)

eth0 Link encap:Ethernet HWaddr fa:16:3e:1c:35:37
 UP BROADCAST RUNNING SLAVE MULTICAST MTU:8888 Metric:1
 RX packets:3236 errors:0 dropped:3177 overruns:0 frame:0
 TX packets:78 errors:0 dropped:0 overruns:0 carrier:0
 collisions:0 txqueuelen:1000
 RX bytes:197273 (197.2 KB) TX bytes:12847 (12.8 KB)

eth1 Link encap:Ethernet HWaddr fa:16:3e:1c:35:37
 UP BROADCAST RUNNING SLAVE MULTICAST MTU:8888 Metric:1
 RX packets:6366 errors:0 dropped:18 overruns:0 frame:0
 TX packets:18224 errors:0 dropped:0 overruns:0 carrier:0
 collisions:0 txqueuelen:1000
 RX bytes:444846 (444.8 KB) TX bytes:1550404 (1.5 MB)

lo Link encap:Local Loopback
 inet addr:127.0.0.1 Mask:255.0.0.0
 inet6 addr: ::1/128 Scope:Host
 UP LOOPBACK RUNNING MTU:65536 Metric:1
```

**步骤 10** 参见上述步骤，完成其他物理机的配置。

**步骤 11** 待其他物理机配置完成后，互相 ping 对端自定义 VLAN 网络配置的同网段 IP，检查是否可以 ping 通。



返回信息示例如下：

```
PS C:\Users\Administrator> Get-NetAdapter

Name InterfaceDescription ifIndex Status MacAddress LinkSpeed
----- -
以太网 6 Intel(R) 82599 10 Gigabit 双端口网络... 18 Up F4-4C-7F-3F-DA-08 10 Gbps
以太网 4 Intel(R) 82599 10 Gigabit 双端口网络... 16 Up F4-4C-7F-3F-DA-07 10 Gbps
eth1 Intel(R) 82599 10 Gigabit 双端口网络... 15 Up E8-4D-D0-C8-99-5C 10 Gbps
eth0 Intel(R) 82599 10 Gigabit 双端口网络... 17 Up E8-4D-D0-C8-99-5B 10 Gbps
Team1 Microsoft Network Adapter Multiplexo... 23 Up FA-16-3E-C8-C4-73 10 Gbps

PS C:\Users\Administrator>
```

## 说明

其中，“eth0”和“eth1”为承载 VPC 网络的网络设备，“以太网 4”和“以太网 6”为承载自定义 VLAN 网络的网络设备。下面步骤将使用“以太网 4”和“以太网 6”配置自定义 VLAN 网络。

**步骤 3** 执行以下命令，创建自定义 VLAN 网络的端口组。

```
New-NetLbfoTeam -Name Team2 -TeamMembers "以太网 4","以太网 6" -TeamingMode
SwitchIndependent -LoadBalancingAlgorithm IPAddresses -Confirm:$false
```

```
PS C:\Users\Administrator> New-NetLbfoTeam -Name Team2 -TeamMembers "以太网 4","以太网 6" -TeamingMode SwitchIndependent
-LoadBalancingAlgorithm IPAddresses -Confirm:$false

Name : Team2
Members : {以太网 4, 以太网 6}
TeamNics : Team2
TeamingMode : SwitchIndependent
LoadBalancingAlgorithm : IPAddresses
Status : Down

PS C:\Users\Administrator>
```

## 说明

其中，“Team2”为给自定义 VLAN 网络端口组规划的端口组名称，“以太网 4”和“以太网 6”为[步骤 2](#)中获取的承载自定义 VLAN 网络的网络设备。

**步骤 4** 执行以下命令，设置[步骤 3](#)中创建的“Team2”端口组中一个网口模式为备用模式。

```
Set-NetLbfoTeamMember -Name "以太网 4" -AdministrativeMode Standby -
Confirm:$false
```

## 说明

当前自定义 VLAN 网络配置的端口组只支持主备模式，其中，“以太网 4”为组成自定义 VLAN 网络端口组中的其中一个端口，配置哪个端口为备用端口，根据您的规划自行决定。

```
PS C:\Users\Administrator> Set-NetLbfoTeamMember -Name "以太网 4" -AdministrativeMode Standby -Confirm:$false
PS C:\Users\Administrator>
```

## get-NetLbfoTeamMember

```
PS C:\Users\Administrator> get-NetLbfoTeamMember

Name : 以太网 4
InterfaceDescription : Intel(R) 82599 10 Gigabit 双端口网络连接 #2
Team : Team2
AdministrativeMode : Standby
OperationalStatus : Standby
TransmitLinkSpeed(Gbps) : 10
ReceiveLinkSpeed(Gbps) : 10
FailureReason : AdministrativeDecision

Name : 以太网 6
InterfaceDescription : Intel(R) 82599 10 Gigabit 双端口网络连接 #4
Team : Team2
AdministrativeMode : Active
OperationalStatus : Active
TransmitLinkSpeed(Gbps) : 10
ReceiveLinkSpeed(Gbps) : 10
FailureReason : NoFailure

Name : eth0
InterfaceDescription : Intel(R) 82599 10 Gigabit 双端口网络连接 #3
Team : Team1
AdministrativeMode : Standby
OperationalStatus : Standby
TransmitLinkSpeed(Gbps) : 10
ReceiveLinkSpeed(Gbps) : 10
FailureReason : AdministrativeDecision

Name : eth1
InterfaceDescription : Intel(R) 82599 10 Gigabit 双端口网络连接
Team : Team1
AdministrativeMode : Active
OperationalStatus : Active
TransmitLinkSpeed(Gbps) : 10
ReceiveLinkSpeed(Gbps) : 10
FailureReason : NoFailure
```

## Get-NetAdapter

```
PS C:\Users\Administrator> Get-NetAdapter

Name InterfaceDescription ifIndex Status MacAddress LinkSpeed
---- -
Team2 Microsoft Network Adapter Multiplexo... 33 Up F4-4C-7F-3F-DA-08 10 Gbps
以太网 6 Intel(R) 82599 10 Gigabit 双端口网络... 18 Up F4-4C-7F-3F-DA-08 10 Gbps
以太网 4 Intel(R) 82599 10 Gigabit 双端口网络... 16 Up F4-4C-7F-3F-DA-07 10 Gbps
eth1 Intel(R) 82599 10 Gigabit 双端口网络... 15 Up E8-4D-D0-C8-99-5C 10 Gbps
eth0 Intel(R) 82599 10 Gigabit 双端口网络... 17 Up E8-4D-D0-C8-99-5B 10 Gbps
Team1 Microsoft Network Adapter Multiplexo... 23 Up FA-16-3E-C8-C4-73 10 Gbps
```

**步骤 5** 执行以下命令，进入“网络连接”界面。

ncpa.cpl

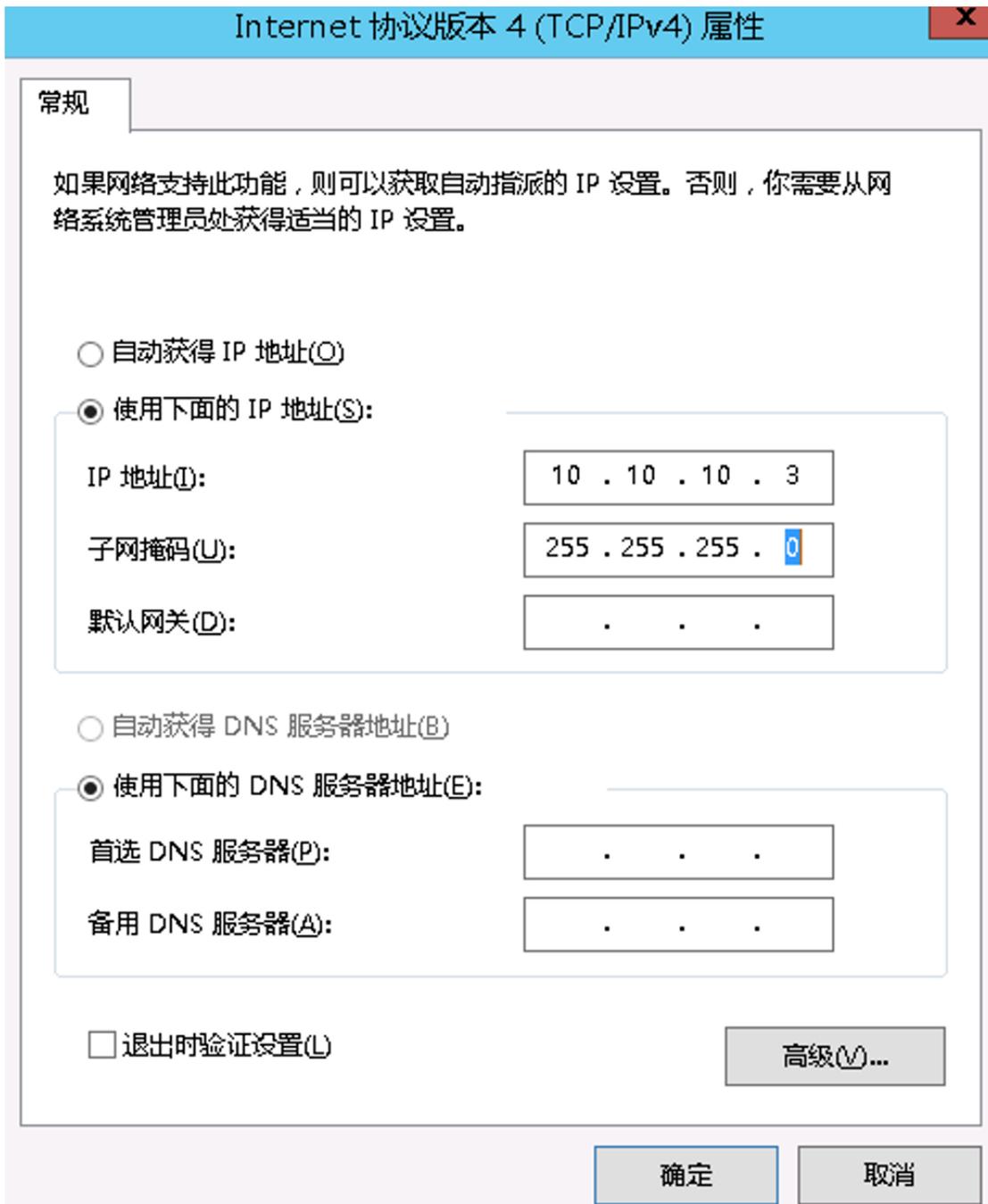
```
PS C:\Users\Administrator> ncpa.cpl
PS C:\Users\Administrator>
```

执行完成后，进入如下界面：



#### 步骤 6 配置自定义 VLAN 网络。

1. 在“网络连接”界面，双击在[步骤 3](#)中创建的端口组“Team2”，进入“Team2 状态”页面。
2. 单击“属性”，进入“Team2 属性”页面。
3. 在“网络”页签下双击“Internet 协议版本 4 (TCP/IPv4)”，进入“Internet 协议版本 4 (TCP/IPv4) 属性”页面。
4. 选择“使用下面的 IP 地址”，配置自定义 VLAN 网络的 IP 地址和子网掩码，单击“确定”。



## 说明

其中，为自定义 VLAN 网络规划的 IP 地址在没有与 VPC 网段冲突的情况下可任意规划，需要通过自定义 VLAN 网络通信的物理机须将自定义 VLAN 网络配置在同一个网段。

**步骤 7** 参见上述步骤，完成其他物理机的配置。

**步骤 8** 待其他物理机配置完成后，互相 ping 对端自定义 VLAN 网络配置的同网段 IP，检查是否可以 ping 通。

```
PS C:\Users\Administrator> ping 10.10.10.4

正在 Ping 10.10.10.4 具有 32 字节的数据:
来自 10.10.10.4 的回复: 字节=32 时间<1ms TTL=128

10.10.10.4 的 Ping 统计信息:
 数据包: 已发送 = 4, 已接收 = 4, 丢失 = 0 (0% 丢失),
 往返行程的估计时间(以毫秒为单位):
 最短 = 0ms, 最长 = 0ms, 平均 = 0ms
PS C:\Users\Administrator>
```

## 4.13 管理权限

### 4.13.1 创建用户并授权使用物理机

本章节主要介绍创建用户并授权使用物理机。

如果您需要对您所拥有的物理机资源进行精细的权限管理，您可以使用统一身份认证服务（Identity and Access Management，简称 IAM），通过 IAM，您可以：

- 根据企业的业务组织，在您的天翼云帐号中，给企业中不同职能部门的员工创建 IAM 用户，让员工拥有唯一安全凭证，并使用 BMS 资源。
- 根据企业用户的职能，设置不同的访问权限，以达到用户之间的权限隔离。
- 将 BMS 资源委托给更专业、高效的其他天翼云帐号或者云服务，这些帐号或者云服务可以根据权限进行代运维。

如果天翼云帐号已经能满足您的要求，不需要创建独立的 IAM 用户，您可以跳过本章节，不影响您使用物理机的其它功能。

操作步骤：

#### 1. 创建用户组并授权

在 IAM 控制台创建用户组，并授予裸金属服务器只读权限“BMS ReadOnlyAccess”。

#### 2. 创建用户并加入用户组

在 IAM 控制台创建用户，并将其加入创建的用户组。

### 3. 用户登录并验证权限

新创建的用户登录控制台，切换至授权区域，验证权限：

- 在“服务列表”中选择“计算 > 物理机服务”，进入物理机主界面，单击右上角“申请物理机”尝试购买物理机。若提示权限不足，表示“BMS ReadOnlyAccess”已生效。
- 在“服务列表”中选择除物理机服务以外的任意一个服务，若提示权限不足，表示“BMS ReadOnlyAccess”已生效。

## 4.13.2 物理机自定义权限策略

本章节主要介绍常用的物理机自定义权限策略。

如果系统预置的 BMS 权限，不满足您的授权要求，可以创建自定义策略。自定义策略中可以添加的授权项（Action）请参见权限及授权项说明。

目前天翼云支持以下两种方式创建自定义策略：

- 可视化视图创建自定义策略：无需了解策略语法，按可视化视图导航栏选择云服务、操作、资源、条件等策略内容，可自动生成策略。
- JSON 视图创建自定义策略：可以在选择策略模板后，根据具体需求编辑策略内容；也可以在编辑框内编写 JSON 格式的策略内容。

### 物理机自定义策略样例

**示例 1：**授权用户修改裸金属服务器名称。

```
{
 "Version": "1.1",
 "Statement": [
 {
 "Effect": "Allow",
 "Action": [
 "bms:servers:list",
 "bms:servers:get",
 "bms:servers:put"
]
 }
]
}
```

**示例 2：**授权用户批量启动服务器。

```
{
 "Version": "1.1",
 "Statement": [
 {
```

```
 "Effect": "Allow",
 "Action": [
 "bms:servers:list",
 "bms:servers:get",
 "bms:servers:start"
]
 }
]
```

**示例 3:** 拒绝用户下电服务器。

拒绝策略需要同时配合其他策略使用，否则没有实际作用。用户被授予的策略中，一个授权项的作用如果同时存在 **Allow** 和 **Deny**，则遵循 **Deny 优先原则**。

如果您给用户授予 **BMS FullAccess** 的系统策略，但不希望用户拥有 **BMS FullAccess** 中定义的下电裸金属服务器权限(**bms:servers:stop**)，您可以创建一条拒绝下电服务器的自定义策略，然后同时将 **BMS FullAccess** 和拒绝策略授予用户，根据 **Deny 优先原则**，则用户可以对 **BMS** 执行除了下电以外的所有操作。以下策略样例表示：拒绝用户下电裸金属服务器。

```
{
 "Version": "1.1",
 "Statement": [
 {
 "Effect": "Deny",
 "Action": [
 "bms:servers:stop"
]
 }
]
}
```

### 4.13.3 自定义策略示例：自定义网络和自定义网络 ACL

本章节仅介绍各场景下自定义网络和自定义网络 ACL 策略的 JSON 文本内容。

自定义网络和自定义网络 ACL 相关策略未在 **BMS FullAccess**、**BMS CommonOperations** 或 **BMS ReadOnlyAccess** 系统策略中定义，您需要创建自定义策略来实现创建、修改、删除自定义网络和自定义网络 ACL 等操作。

#### 场景一：自定义网络和自定义网络 ACL 依赖的授权项

自定义网络和自定义网络 ACL 依赖的授权项必须包含：**ecs:servers:list**、**bms:servers:list**

```
{
 "Version": "1.1",
 "Statement": [
 {
 "Effect": "Allow",
```

```
 "Action": [
 "ecs:servers:list",
 "bms:servers:list"
]
 }
]
```

如果未添加这些授权项，用户将无法进入裸金属服务器列表页面，也就无法进行任何自定义网络和自定义网络 ACL 相关的操作。

## 场景二：创建自定义网络

创建自定义网络对应授权项为：`bms:virtualNetworks:create`。

除了依赖[场景一：自定义网络和自定义网络 ACL 依赖的授权项](#)中的授权项外，还依赖 `vpc:vpcs:list`，因为自定义网络创建页面会查询 VPC 列表。

完整的策略内容如下：

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

## 场景三：查询自定义网络列表

查询自定义网络列表对应授权项为：`bms:virtualNetworks:list`

完整的策略内容如下

```
{
 "Version": "1.1",
 "Statement": [
 {
 "Effect": "Allow",
 "Action": [
 "ecs:servers:list",
 "bms:servers:list",
 "vpc:vpcs:list",
 "bms:virtualNetworks:list"
]
 }
]
}
```

```
]
}
```

#### 场景四：查询自定义网络详情

查询自定义网络详情对应授权项为：`bms:virtualNetworks:get`

完整的策略内容如下：

```
{
 "Version": "1.1",
 "Statement": [
 {
 "Effect": "Allow",
 "Action": [
 "ecs:servers:list",
 "bms:servers:list",
 "vpc:vpcs:list",
 "bms:virtualNetworks:list",
 "bms:virtualNetworks:get"
]
 }
]
}
```

#### 场景五：修改自定义网络名称

修改自定义网络名称对应授权项为：`bms:virtualNetworks:update`

完整的策略内容如下：

```
{
 "Version": "1.1",
 "Statement": [
 {
 "Effect": "Allow",
 "Action": [
 "ecs:servers:list",
 "bms:servers:list",
 "vpc:vpcs:list",
 "bms:virtualNetworks:list",
 "bms:virtualNetworks:get",
 "bms:virtualSubnets:create",
 "bms:virtualNetworks:update"
]
 }
]
}
```

#### 场景六：删除自定义网络

删除自定义网络对应授权项为：`bms:virtualNetworks:delete`

完整的策略内容如下：

```
{
 "Version": "1.1",
 "Statement": [
 {
 "Effect": "Allow",
 "Action": [
 "ecs:servers:list",
 "bms:servers:list",
 "vpc:vpcs:list",
 "bms:virtualNetworks:list",
 "bms:virtualNetworks:get",
 "bms:virtualNetworks:delete"
]
 }
]
}
```

### 场景七：添加自定义子网

添加自定义子网对应授权项为：**bms:virtualSubnets:create**

完整的策略内容如下：

```
{
 "Version": "1.1",
 "Statement": [
 {
 "Effect": "Allow",
 "Action": [
 "ecs:servers:list",
 "bms:servers:list",
 "vpc:vpcs:list",
 "bms:virtualNetworks:list",
 "bms:virtualNetworks:get",
 "bms:virtualSubnets:list",
 "bms:virtualSubnets:create"
]
 }
]
}
```

### 场景八：查询自定义子网列表

查询自定义子网列表对应授权项为：**bms:virtualSubnets:list**

完整的策略内容如下：

```
{
 "Version": "1.1",
 "Statement": [
 {
 "Effect": "Allow",
```

```
 "Action": [
 "ecs:servers:list",
 "bms:servers:list",
 "vpc:vpcs:list",
 "bms:virtualNetworks:list",
 "bms:virtualNetworks:get",
 "bms:virtualSubnets:list"
]
 }
]
```

### 说明:

该授权项仅用于自定义网络 ACL 关联自定义子网时使用。

### 场景九：删除自定义子网

删除自定义子网对应授权项为：**bms:virtualSubnets:delete**

完整的策略内容如下：

```
{
 "Version": "1.1",
 "Statement": [
 {
 "Effect": "Allow",
 "Action": [
 "ecs:servers:list",
 "bms:servers:list",
 "vpc:vpcs:list",
 "bms:virtualNetworks:list",
 "bms:virtualNetworks:get",
 "bms:virtualSubnets:list",
 "bms:virtualSubnets:delete"
]
 }
]
}
```

### 场景十：创建自定义网络 ACL

创建自定义网络 ACL 对应授权项为：**bms:firewallGroups:create**

完整的策略内容如下：

```
{
 "Version": "1.1",
 "Statement": [
 {
 "Effect": "Allow",
 "Action": [
 "ecs:servers:list",
```

```

 "bms:servers:list",
 "vpc:vpcs:list",
 "bms:firewallGroups:list",
 "bms:firewallGroups:create"
]
}

```

### 场景十一：查询自定义网络 ACL 列表

查询自定义网络 ACL 列表对应授权项为：`bms:firewallGroups:list`

完整的策略内容如下：

```

{
 "Version": "1.1",
 "Statement": [
 {
 "Effect": "Allow",
 "Action": [
 "ecs:servers:list",
 "bms:servers:list",
 "vpc:vpcs:list",
 "bms:firewallGroups:list"
]
 }
]
}

```

### 场景十二：查询自定义网络 ACL 详情

查询自定义网络 ACL 详情对应授权项为：`bms:firewallGroups:get`

完整的策略内容如下：

```

{
 "Version": "1.1",
 "Statement": [
 {
 "Effect": "Allow",
 "Action": [
 "ecs:servers:list",
 "bms:servers:list",
 "vpc:vpcs:list",
 "bms:firewallGroups:list",
 "bms:firewallGroups:get"
]
 }
]
}

```

### 场景十三：修改自定义网络 ACL

该场景包括如下操作：修改名称、修改描述、添加 ACL 规则、修改 ACL 规则、删除 ACL 规则、开启/关闭 ACL 规则、向前/后插入规则、关联自定义子网（依赖 `bms:virtualSubnets:list` 授权项）。

修改自定义网络 ACL 对应授权项为：`bms:firewallGroups:update`

完整的策略内容如下：

```
{
 "Version": "1.1",
 "Statement": [
 {
 "Effect": "Allow",
 "Action": [
 "ecs:servers:list",
 "bms:servers:list",
 "vpc:vpcs:list",
 "bms:firewallGroups:list",
 "bms:firewallGroups:get",
 "bms:virtualSubnets:list",
 "bms:firewallGroups:update"
]
 }
]
}
```

#### 场景十四：删除自定义网络 ACL

删除自定义网络 ACL 对应授权项为：`bms:firewallGroups:delete`

完整的策略内容如下：

```
{
 "Version": "1.1",
 "Statement": [
 {
 "Effect": "Allow",
 "Action": [
 "ecs:servers:list",
 "bms:servers:list",
 "vpc:vpcs:list",
 "bms:firewallGroups:list",
 "bms:firewallGroups:get",
 "bms:firewallGroups:delete"
]
 }
]
}
```

## 4.14 资源与标签

### 4.14.1 标签

#### 4.14.1.1 标签概述

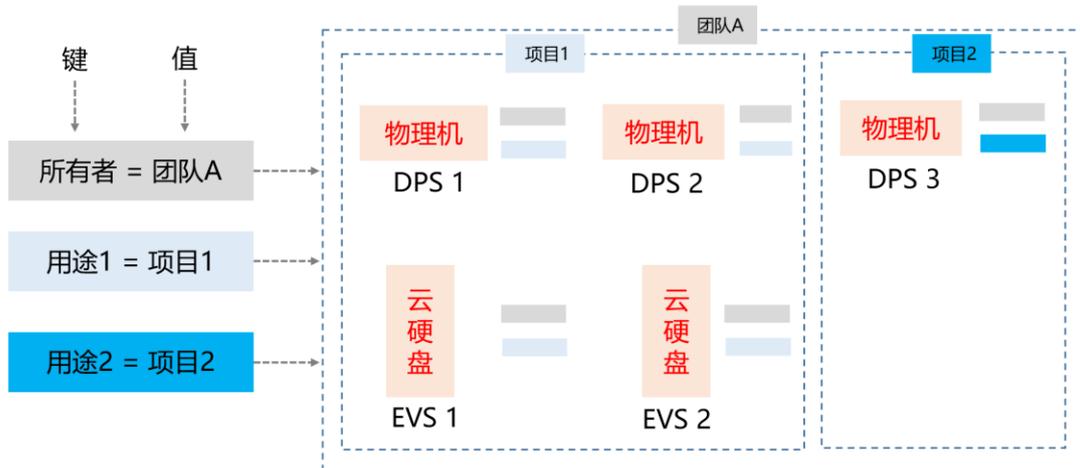
本章节主要介绍标签概念。

为了方便管理您的实例、磁盘、镜像以及其他云资源，您可以通过标签的形式为每个资源分配您自己的元数据。标签管理服务（Tag Management Service，TMS）是一种快速便捷将标签集中管理的可视化服务，提供跨区域、跨服务的集中标签管理和资源分类功能。

#### 标签的基本知识

标签用于标识资源，当您拥有相同类型的许多云资源时，可以使用标签按各种维度（例如用途、所有者或环境）对云资源进行分类。

标签示例图：



上图说明了标签的工作方式。在此示例中，您为每个云资源分配了两个标签，每个标签都包含您定义的一个“键”和一个“值”，一个标签使用键为“所有者”，另一个使用键为“用途”，每个标签都拥有相关的值。

您可以根据为云资源添加的标签快速搜索和筛选特定的云资源。例如，您可以为帐户中的资源定义一组标签，以跟踪每个云资源的所有者和用途，使资源管理变得更加轻松。

#### 标签使用说明：

- 支持添加标签的物理机相关服务有：物理机、弹性云主机、镜像服务、云硬盘等。
- 每个标签由一对键值对（Key-Value）组成。
- 每个物理机最多可以添加 9 个标签。
- 对于每个资源，每个标签键（Key）都必须是唯一的，每个标签键（Key）只能有一个值（Value）。
- 标签命名规则 如下图所示

### 标签命名规则：

| 参数  | 规则                                                                                                                                  | 样例           |
|-----|-------------------------------------------------------------------------------------------------------------------------------------|--------------|
| 标签键 | <ul style="list-style-type: none"> <li>• 不能为空。</li> <li>• 只能包含英文字母、数字、下划线（_）、中划线（-）、中文字符。</li> <li>• 长度最大为 36 个字符。</li> </ul>       | Organization |
| 标签值 | <ul style="list-style-type: none"> <li>• 不能为空。</li> <li>• 只能包含英文字母、数字、下划线（_）、点号（.）、中划线（-）、中文字符。</li> <li>• 长度最大为 43 个字符。</li> </ul> | Apache       |

#### 4.14.1.2 添加标签

本章节主要介绍添加标签。

标签用于标记云资源，如实例、镜像和磁盘等。如果您的帐户下有多种云资源，并且不同云资源之间有多种关联，您可以为云资源添加标签，实现云资源的分类和统一管理

#### 在创建物理机时添加标签

1. 登录管理控制台。
2. 选择“计算 > 物理机服务”。  
进入物理机页面。
3. 单击“申请物理机”。
4. 配置物理机的规格参数。
5. 在“高级配置 > 标签”中，添加标签键和标签值。

高级配置 暂不配置 现在配置

标签 您可以选择已有的标签，或自定义新的标签。 C

用途1  项目1  删除

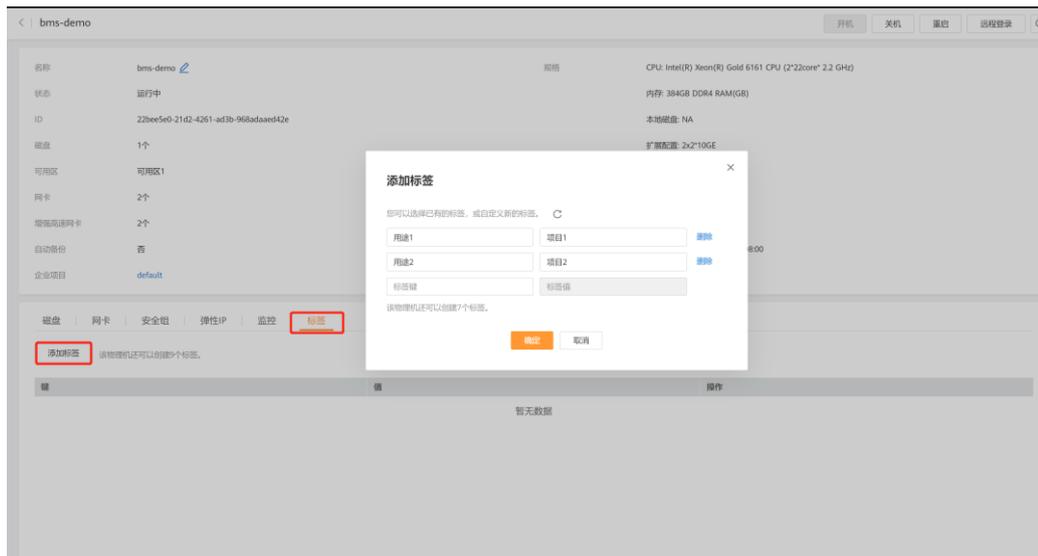
标签键  标签值

您还可以增加8个标签

委托 ? --请选择-- C 新建委托

### 在物理机详情页添加标签：

1. 登录管理控制台。
2. 选择“计算 > 物理机服务”。  
进入物理机页面。
3. 在物理机列表中，单击待添加标签的物理机名称。  
系统跳转至该物理机详情页面。
4. 选择“标签”页签，单击“添加标签”，在弹出的“添加标签”窗口，输入标签的键和值。



您还可以在标签添加成功后，执行修改操作，只能修改标签值。

#### 4.14.1.3 使用标签检索资源

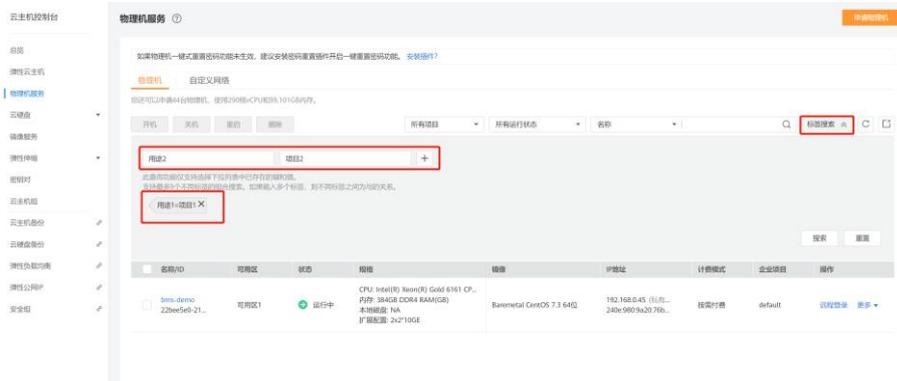
本章节主要介绍使用标签检索资源。

为云资源添加标签后，您可以通过本文所述的两种方法使用标签检索资源。

#### 在资源列表中筛选资源

在物理机列表页，您可以按“标签键+标签值”搜索目标物理机。

1. 登录管理控制台。
2. 选择“计算 > 物理机”。  
进入物理机页面。
3. 单击物理机列表右上角的“标签搜索”，展开查询页。
4. 输入待查询物理机的标签。  
标签键和标签值均不能为空，当标签键和标签值全匹配时，系统可以自动查询到目标物理机。
5. 单击标签加号添加多个标签。  
系统支持不同标签的组合搜索，并取各个标签的交集。
6. 单击“搜索”。  
系统根据标签键或标签值搜索目标物理机。



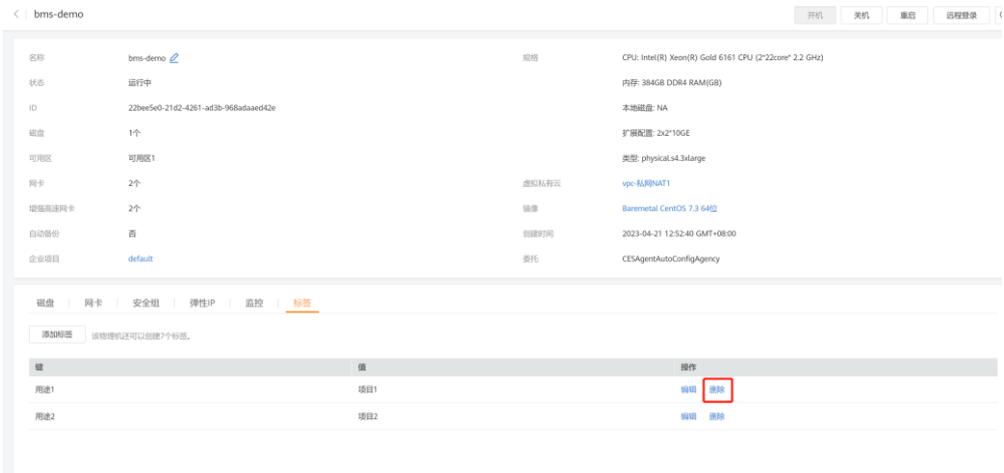
#### 4.14.1.4 删除标签

本章节主要介绍删除标签。

如果某个标签已经不再适用于您的资源管理，您可以删除资源标签。

##### 操作步骤

1. 登录管理控制台。
2. 选择“计算 > 物理机”。  
进入物理机页面。
3. 在物理机列表中，单击待删除标签的物理机名称。  
系统跳转至该物理机详情页面。
4. 选择“标签”页签，单击标签所在行“操作”列下的“删除”，如果确认删除，在弹出的“删除标签”窗口，单击“确定”



## 4.14.2 资源位置

有些资源可以在所有地区（全球）使用，而有些资源则特定于其所在的区域或可用区。

| 资源         | 类型      | 描述                                                                                              |
|------------|---------|-------------------------------------------------------------------------------------------------|
| 帐户         | 全球性     | 您可以在所有区域使用同一个帐户。                                                                                |
| 预定义标签      | 全球性     | 您可以在所有区域使用同一个预定义标签。                                                                             |
| 密钥对        | 全球性或区域性 | 您在管理控制台创建的密钥对与您在其中创建它们的区域相关联。您可以创建自己的 RSA 密钥对并将其导入到您打算在其中使用它的区域。因此，您可以通过将密钥对上传至每个区域而使其在全球范围内使用。 |
| 资源标识符      | 区域性     | 每个资源的标识符（例如，实例 ID、云硬盘 ID、虚拟私有云 ID）都与其区域相关联，并且只能在创建资源的区域使用。                                      |
| 用户自定义的资源名称 | 区域性     | 每个资源名称（例如，安全组名称、密钥对名称）都与其区域相关联，并且只能在创建资源的区域使用。尽管您可以在多个区域创建名称相同的资源，但是它们之间并无关联。                   |
| 虚拟私有云      | 区域性     | 虚拟私有云与区域相关联，并且只能与同一区域的实例相关联。                                                                    |
| 弹性公网 IP    | 区域性     | 弹性公网 IP 与区域相关联，并且只能与同一区域的实例相关联。                                                                 |
| 安全组        | 区域性     | 安全组与区域相关联，并且只能分配给同一区域的实例。不能通过安全组规则实现一个实例与其所在区域外的实例通信。                                           |

|    |     |                                                 |
|----|-----|-------------------------------------------------|
| 镜像 | 区域性 | 镜像与区域相关联，并且只能与同一区域的实例相关联。这里的镜像包括公共镜像、私有镜像、共享镜像。 |
| 实例 | 可用区 | 实例与可用区相关联，但是实例 ID 与区域相关联。                       |
| 磁盘 | 可用区 | 磁盘与可用区相关联，只能挂载到同一可用区的实例上。                       |
| 子网 | 可用区 | 子网与可用区相关联，只能与同一可用区的实例相关联。                       |

## 4.15 物理机监控

### 4.15.1 物理机监控概述

本章节主要描述了物理机的监控。

云监控服务的主机监控分为基础监控和操作系统监控，基础监控为云服务器自动上报的监控指标（物理机服务器不支持基础监控）。操作系统监控通过在物理机服务器中安装 Agent 插件，为用户提供服务器的系统级、主动式监控服务。Agent 占用的系统资源很小，CPU 用率<1.5%、内存<50M。

云监控会提供 CPU、内存、磁盘、网络等四十余种监控指标，满足服务器的基本监控运维需求。

#### 约束与限制

- 用户私有镜像不在支持范围内。
- Agent 插件目前仅支持 64 位 Linux 操作系统的物理机服务器实例。  
主机监控对 Linux 镜像支持列表如下表所示。

#### 主机监控对 Linux 镜像支持列表

| 操作系统类型       | 支持版本号                        |
|--------------|------------------------------|
| Red Hat      | 6.5, 6.7, 6.8, 7.2, 7.3, 7.4 |
| SUSE         | 11.4, 12.1                   |
| Oracle Linux | 6.5, 7.3, 7.4                |

| 操作系统类型 | 支持版本号              |
|--------|--------------------|
| CentOS | 6.9, 7.2, 7.3, 7.4 |

## 4.15.2 安装配置 Agent

本章节主要介绍安装配置物理机监控 agent。

### 4.15.2.1 操作场景

当您购买了一台物理机后，了解其运行状态一定是您的迫切需求，天翼云物理机和云监控服务结合使用，自动收集物理机的 CPU、内存、磁盘以及网络使用情况等监控指标，以便您及时了解物理机实例运行状况和性能。本章节指导您如何为物理机服务器安装及配置 Agent 插件。

### 4.15.2.2 前提条件

- 物理机服务器可以正常使用。
- 物理机服务器内 DNS Server 正常工作，配置方法请参考[添加 DNS 服务解析配置](#)。

### 4.15.2.3 操作步骤

1. 添加域名解析地址：在物理机“/etc/resolv.conf”文件中添加各区域域名解析地址。
2. 修改子网 DNS 地址：按物理机所在区域修改子网的 DNS 服务器地址。
3. 配置安全组：用于下载 Telescope 包、发送指标数据、采集日志等。  
此第三步请参见安装配置[管理 Agent](#)中添加 DNS 服务解析处理
4. 下载 telescope 安装包，访问地址参考[云监控服务-用户指南-安装 Agent](#)。
5. 安装 Agent：手动为物理机安装 Agent，实现主机监控。  
此步请参见安装配置[安装 Agent \(Linux\)](#)中安装 Agent 处理。
6. 配置 Agent，使用 root 账号，登录物理机。
7. 执行以下命令，切换至 Agent 安装路径的 bin 下。

```
cd /usr/local/telescope/bin
```

8. 修改配置文件 conf.json。

- a. 执行以下命令，打开配置文件 conf.json。

```
vi conf.json
```

- b. 修改文件中的参数，具体参数请参见下表。

```
{
 "InstanceId": "xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx",
 "ProjectId": "b5b92ee0xxxxxxxxxxxxxxxxcab92396",
 "AccessKey": "QZ0XGJXFxxxxxxxxT65R",
 "SecretKey": "lEv2aXAGxxxxxxxxxxxxxxxxxxxxF8t0Bf18Tn2",
 "RegionId": "cn-hz1"
}
```

公共配置参数

| 参数         | 说明                                                                                                                                                                                                                                                      |
|------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| InstanceId | <p>物理机 ID，可通过登录管理控制台，在物理机列表中查看。</p> <p>说明</p> <p>InstanceId 可不用配置，保持"InstanceId":""，即可，若需要配置，请参考下两条。</p> <p>该资源 ID 需保证全局唯一性，即同一个 RegionID 下 Agent 使用的 InstanceId 不能相同，否则系统可能会出现异常。</p> <p>InstanceId 必须与实际的物理机资源 ID 一致，否则云监控界面将看不到对应物理机资源操作系统监控的数据。</p> |
| ProjectId  | <p>ProjectId 可不用配置，保持"ProjectId": ""，即可。</p> <p>若需要配置，请参考已下获取方式。</p> <p>项目 ID，获取方式如下：</p>                                                                                                                                                               |

| 参数                  | 说明                                                                                                                                                                                                                                                                                                                                                                            |
|---------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|                     | <ol style="list-style-type: none"> <li>1. 登录管理控制台，单击右上角“用户名”，选择“我的凭证”；</li> <li>2. 在项目列表中，查看物理机资源对应的所属区域的项目 ID。</li> </ol>                                                                                                                                                                                                                                                    |
| AccessKey/SecretKey | <p>访问密钥，获取方式如下：</p> <p>登录管理控制台，单击右上角“用户名”，选择“我的凭证&gt;管理访问密钥”；</p> <p>如已有访问密钥，查看创建时下载保存的 credentials.csv 文件中，获取文件中记录的 Key 值即可；</p> <p>如未创建，则通过“新增访问密钥”可创建新的访问密钥，妥善保存 credentials.csv 文件，并获取文件中记录的 Key 值。</p> <p><b>注意</b></p> <p>为了安全考虑，建议该用户为 IAM 用户，并且权限仅为 CES Administrator 和 LTS Administrator。</p> <p>配置的 AccessKey 必须在“我的凭证 &gt;管理访问密钥”列表中，否则将鉴权失败，云监控界面看不到操作系统监控数据。</p> |
| RegionId            | <p>区域 ID，例如：物理机资源所属区域为“杭州”，则 RegionID 为“cn-hz1”。</p>                                                                                                                                                                                                                                                                                                                          |

9. 修改云监控指标采集模块的配置文件 conf\_ces.json。

- a. 执行以下命令，打开公共配置文件 conf\_ces.json。

```
vi conf_ces.json
```

- b. 修改文件中的参数，具体参数请参见下表。

```
{
```

```
 "Endpoint": "https://ces.cn-hz1.ctyun.cn"
```

```
}

```

- 指标采集模块参数配置

| 参数              | 说明                                                                           |
|-----------------|------------------------------------------------------------------------------|
| <b>Endpoint</b> | 物理机资源所属区域的云监控 Endpoint URL，例如：物理机资源所属区域为“杭州”，则 URL 中使用“ces.cn-hzl.ctyun.cn”。 |

说明：

- Agent 插件配置完成后，因监控数据暂未上报，插件状态仍显示“未安装”，等待 3-5 分钟，刷新即可。

10. 管理 Agent ，查看 Agent 状态。
11. 登录天翼云物理机服务器，执行以下命令，查看 Agent 状态。

```
service telescoped status
```

当系统返回以下内容，则表示 Agent 为正常运行状态。

```
"Telescope process is running well."
```

### 启动 Agent

执行以下命令，启动 Agent。

```
/usr/local/telescope/telescoped start
```

### 重启 Agent

执行以下命令，重启 Agent。

```
/usr/local/telescope/telescoped restart
```

### 停止 Agent

执行以下命令，停止 Agent。

```
service telescoped stop
```

### 说明：

如果 Telescope 安装失败，可能会导致无法正常停止 Agent，可通过执行以下命令进一步尝试：

```
/usr/local/telescope/telescoped stop
```

### 卸载 Agent

用户可手动卸载 Agent 插件，卸载后将不再监控 BMS 实例监控数据。如需再次使用，请参考安装 Agent 重新安装。

执行以下命令，即可卸载 Agent。

```
/usr/local/telescope/uninstall.sh
```

### 12. 查看监控指标

以上配置完成后，进入控制台界面，选择“管理与部署 > 云监控”，在左侧导航栏选择“主机监控 > 物理机”，列表展示该物理机的名称/ID、主机状态、插件状态等信息。

## 4.15.3 物理机支持的监控指标

本章节主要介绍物理机支持的监控指标。

物理机服务器支持的监控指标有：CPU 相关监控指标、内存相关监控指标、磁盘相关监控指标、网络和进程相关监控指标，具体如下表所示：

物理机支持的监控指标：

| 指标名称            | 指标含义                          | 取值范围            | 备注       |
|-----------------|-------------------------------|-----------------|----------|
| CPU使用率 (%)      | 该指标用于统计测量对象的 CPU 使用率，以百分比为单位。 | 0 -100 %        | 测量对象：物理机 |
| 内存使用率 (%)       | 该指标用于统计测量对象的内存使用率，以百分比为单位。    | 0 -100 %        | 测量对象：物理机 |
| 磁盘使用率 (%)       | 该指标用于统计测量对象的磁盘使用率，以百分比为单位。    | 0 -100 %        | 测量对象：物理机 |
| 磁盘读速率<br>(字节/秒) | 该指标用于统计每秒从测量对象读出数据量，以字节/秒为单位。 | $\geq 0$ Byte/s | 测量对象：物理机 |

|                             |                                      |                    |          |
|-----------------------------|--------------------------------------|--------------------|----------|
| 磁盘写速率<br>(字节/秒)             | 该指标用于统计每秒写到测量对象的数据量，以字节/秒为单位。        | $\geq 0$ Byte/s    | 测量对象：物理机 |
| 磁盘读操作速率 (请求/秒)              | 该指标用于统计每秒从测量对象读取数据的请求次数，以请求/秒为单位。    | $\geq 0$           | 测量对象：物理机 |
| 磁盘写操作速率 (请求/秒)              | 该指标用于统计每秒从测量对象写数据的请求次数，以请求/秒为单位。     | $\geq 0$           | 测量对象：物理机 |
| 网络流入速率<br>(字节/秒)            | 该指标用于在测量对象内统计每秒流入测量对象的网络流量，以字节/秒为单位。 | $\geq 0$ Byte/s    | 测量对象：物理机 |
| 网络流出速率<br>(字节/秒)            | 该指标用于在测量对象内统计每秒流出测量对象的网络流量，以字节/秒为单位。 | $\geq 0$ Byte/s    | 测量对象：物理机 |
| 网络包流入速率 (包/秒)               | 该指标用于统计每秒流入测量对象的网络包流量，以包/秒为单位。       | $\geq 0$ Packets/s | 测量对象：物理机 |
| 网络包流出速率 (包/秒)               | 该指标用于统计每秒流出测量对象的网络包流量，以包/秒为单位。       | $\geq 0$ Packets/s | 测量对象：物理机 |
| 1 分钟内 CPU 负载<br>(Tasks/CPU) | 该指标用于统计在过去 1 分钟内测量对象的 CPU 负载情况。      | $\geq 0$           | 测量对象：物理机 |

|                              |                                  |          |          |
|------------------------------|----------------------------------|----------|----------|
| 5 分钟内 CPU 负载<br>(Tasks/CPU)  | 该指标用于统计在过去 5 分钟内测量对象的 CPU 负载情况。  | $\geq 0$ | 测量对象：物理机 |
| 15 分钟内 CPU 负载<br>(Tasks/CPU) | 该指标用于统计在过去 15 分钟内测量对象的 CPU 负载情况。 | $\geq 0$ | 测量对象：物理机 |

## 4.16 故障排除

本章节主要介绍物理机故障排除方式。

### 4.16.1 物理机开机或者重启后，无法登录或 EVS 磁盘丢失，如何解决？

物理机开机或者重启后，无法登录或 EVS 磁盘丢失，如何解决？

#### 问题描述

物理机开机或重启后，无法登录或 EVS 磁盘丢失。

#### 可能原因

网络拥塞或者丢包导致物理机获取 IP 或者挂载数据卷（EVS 磁盘）失败。

#### 解决方案

重启物理机，如果多次重启无效，请联系客服。

### 4.16.2 通过 puttygen.exe 工具创建的密钥对，导入管理控制台失败怎么办？

通过 puttygen.exe 工具创建的密钥对，导入管理控制台失败怎么办？

## 问题描述

通过 puttygen.exe 工具创建的密钥对，在导入管理控制台使用时，系统提示导入公钥文件失败。

## 可能原因

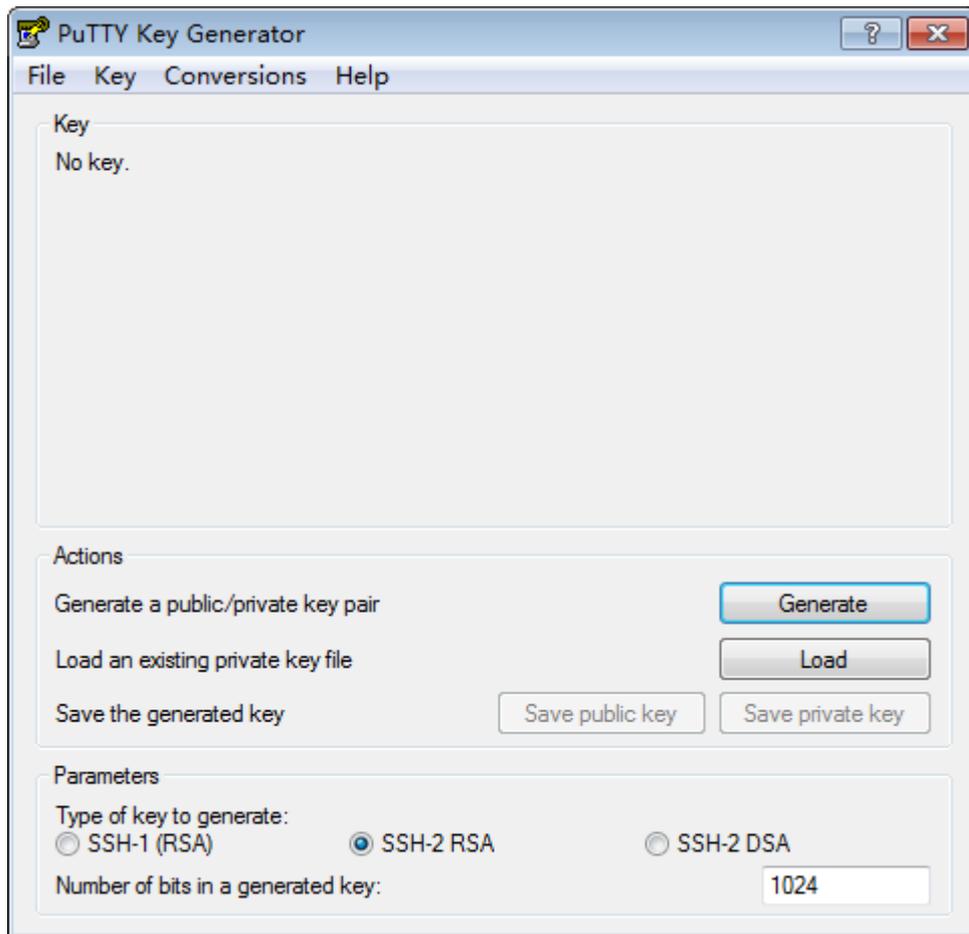
公钥内容的格式不符合系统要求。

当用户使用 puttygen.exe 工具创建密钥对时，如果使用 puttygen.exe 工具的“Save public key”按钮保存公钥，公钥内容的格式会发生变化，不能直接导入管理控制台使用。

## 处理方法

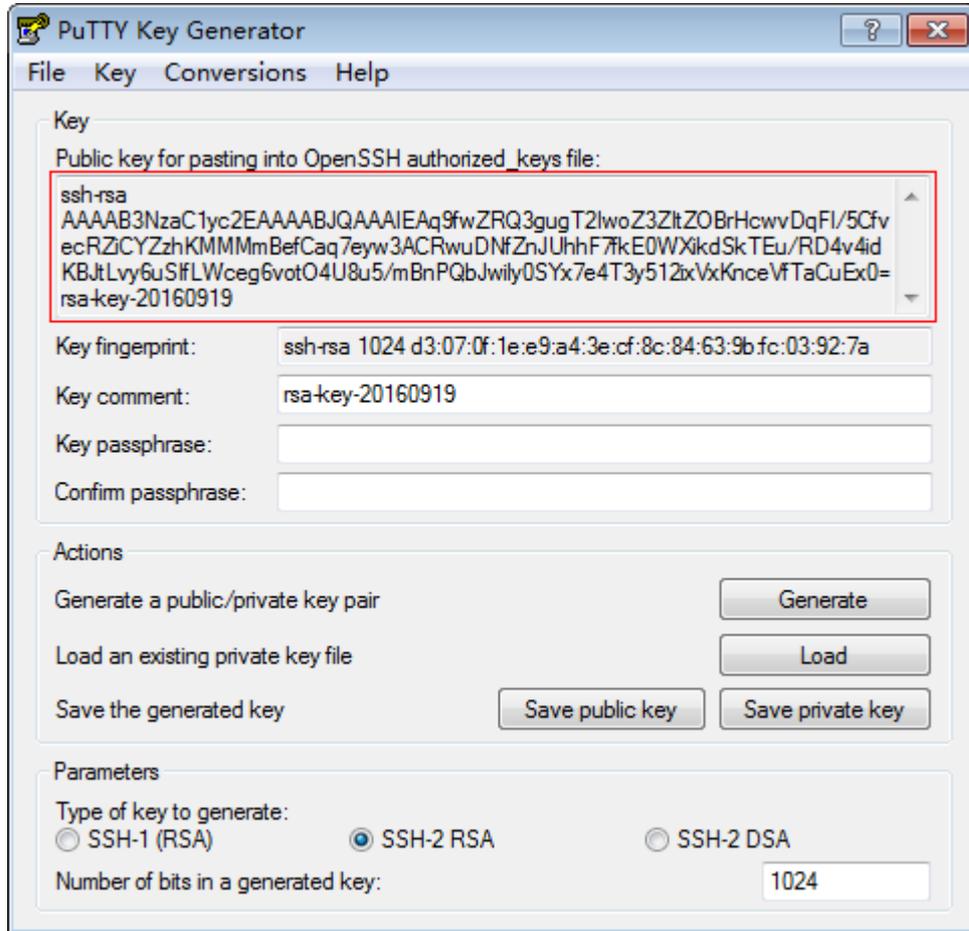
使用本地保存的私钥文件，在“PuTTY Key Generator”中恢复内容格式正确的公钥文件，然后再将该公钥文件导入管理控制台。

1. 双击“PUTTYGEN.EXE”，打开“PuTTY Key Generator”。



2. 单击“Load”，并在本地选择该密钥对的私钥文件。

系统将自动加载该私钥文件，并在“PuTTY Key Generator”中恢复格式正确的公钥文件内容，如下图所示，红框中的内容即为符合系统要求的公钥文件。



3. 复制红框中的公钥内容，并将其粘贴在文本文档中，以 .txt 格式保存在本地，保存公钥文件。
4. 将公钥文件导入管理控制台。
  - a. 登录管理控制台。
  - b. 单击管理控制台左上角的 ，选择地域和项目。
  - c. 选择“计算 > 物理机”。
  - d. 在左侧导航树中，选择“密钥对”。
  - e. 在“密钥对”页面，单击“导入密钥对”。
  - f. 将“.txt”格式文本文档中的公钥内容粘贴至“Public Key Content”的空白区域，并单击“OK”，导入公钥文件。

### 4.16.3 物理机异常重启后无法挂卷，如何处理？

物理机异常重启后无法挂卷，如何处理？

#### 问题现象

携数据卷的本地盘发放物理机，异常重启后发现操作系统内没有卷信息，也无法在管理控制台上挂卷。

这里的异常重启指的是除租户在控制台上操作之外的其他非正常上下电。

#### 解决方案

在待处理的物理机所在行，单击“更多 > 重启”，重启之后卷就会自动挂载。

如果重启后仍无法挂卷，请联系客服。

### 4.16.4 Windows 物理机挂载云硬盘后提示脱机，如何解决？

#### 问题描述

Windows 物理机挂载云硬盘后，打开“控制面板”，选择其中的“系统和安全”，选择“管理工具”，双击“计算机管理”。在“计算机管理”页面选择“存储 > 磁盘管理”，查看挂载的云硬盘显示脱机状态。

#### 解决方案

1. 登录 Windows 物理机操作系统。
2. 单击“开始”菜单，在“搜索程序和文件”中输入 `cmd` 并按“Enter”，打开命令提示符窗口。

3. 输入命令 `diskpart`。

```
C:\Users\Administrator>diskpart
```

4. 输入命令 `san`。

```
DISKPART> san
```

SAN 策略： 全部联机

5. 输入命令 `san policy=onlineall`。

```
DISKPART> san policy=onlineall
```

DiskPart 已成功更改用于当前操作系统的 SAN 策略。

6. 输入命令 `list disk`，将显示物理机所有磁盘信息。

```
DISKPART> list disk
```

| 磁盘 | ### | 状态 | 大小     | 可用     | Dyn | Gpt |
|----|-----|----|--------|--------|-----|-----|
| 磁盘 | 0   | 联机 | 838 GB | 0B     |     |     |
| 磁盘 | 1   | 脱机 | 838 GB | 838 GB |     |     |
| 磁盘 | 2   | 脱机 | 838 GB | 838 GB |     |     |
| 磁盘 | 3   | 脱机 | 838 GB | 838 GB |     |     |

...

7. 输入命令 `select disk num`。其中，`num` 表示磁盘号，执行命令时要替换为具体的磁盘号。

```
DISKPART> select disk 4
```

8. 输入命令 `attributes disk clear readonly`。

```
DISKPART> attributes disk clear readonly
```

已成功清除磁盘属性。

9. 输入命令 `online disk`。

```
DISKPART> online disk
```

DiskPart 成功使所选磁盘联机。

10. 修改完成后即可进行格式化云硬盘。

# 5 常见问题

## 常见问题

### 5.1.1 通用类

本章节主要介绍了物理机通用类的问题。

#### 5.1.1.1 物理机服务器有哪些限制？

- 不支持直接加载外接硬件设备（如 USB 设备、银行 U key、外接硬盘、加密狗等）。
- 不支持带外管理，您的物理机资源统一由天翼云管理和维护。
- 不支持热迁移，服务器故障后会对业务造成影响，建议您通过业务集群部署、主备部署等方式实现业务的高可用。
- 不支持创建没有操作系统的裸设备，即物理机必须自带操作系统。
- 不支持更换物理机的操作系统。
- 创建后不支持更换 VPC。
- 公共镜像的 Windows 物理机系统是默认激活的，您上面二次虚拟化出来的操作系统，需要您自行激活。
- 不支持自定义物理机 CPU 和内存配置，不支持规格变配，仅支持云硬盘在线扩容。
- 仅支持挂载 SCSI 类型的云硬盘。
- 由于某些机型的服务器没有配备 SDI 卡，或者其他服务器本身的原因，有些规格或镜像的物理机不支持挂载云硬盘。
- 请勿删除或者修改镜像中内置的插件服务（如 Cloud-Init、bms-network-config 等），否则会影响您的基本功能使用。
- 在创建物理机时，如果选择自动分配 IP 地址，请不要在物理机发放完成后修改私有 IP 地址，避免和其他物理机 IP 冲突。
- 物理机不支持配置桥接网卡，会导致网络不通。

- 禁止升级 OS 自带内核版本，否则服务器硬件驱动会存在兼容性风险，影响服务器可靠性。
- 物理机禁止修改和配置系统盘。

### 5.1.1.2 天翼云物理机服务与传统物理机有什么区别？

**天翼云物理机：**与公有云产品无缝连接，高可用、低延时、易扩展；逻辑隔离的专用网络，更安全；监控可视化、快速发放、VPC 互联；支持挂载共享云硬盘，按需支付实例费用，免操作系统安装，免运维，无需投入大量资金。可以像虚拟机一样灵活的发放和使用，同时又具备了优秀的计算、存储、网络能力。天翼云物理机服务器具备物理机的一切特性和优势。

**传统 IDC 物理机：**自建物理机功能单一，扩展性差，需投入大量人力成本和资源成本，包括服务器、系统、数据库等软硬件费用，及机房费用和运维成本等等。

### 5.1.1.3 天翼云物理机服务器支持哪些系统镜像？

天翼云物理机支持 Linux 和 Windows 主流镜像版本。

支持的 Linux 镜像包括：

- RedHat Linux Enterprise 6.8/6.9/7.2/7.3/7.4/7.5 64 位
- CentOS 6.8/6.9/7.2/7.3/7.4 64 位
- SUSE Linux Enterprise Server 11.4/12.1/12.2/12.3 64 位
- Oracle Linux Server Release 6.9/7.3 64 位
- Ubuntu 16.04 64 位

支持的 Windows 镜像包括：

- 2012 R2 Standard 64 位

需要注意的是，不同规格的物理机服务器支持的操作系统版本有所差异，详情请参见[物理机服务器类型与支持的操作系统版本](#)

### 5.1.1.4 天翼云物理机服务器如何保证数据安全？

- 天翼云物理机服务器具备物理机级的性能和隔离性，用户独占计算资源，并且无任何虚拟化开销。存储在高性能高可靠的服务器上的数据，自然也会很安全。

- 带有本地磁盘的物理机服务器，支持本地磁盘组 RAID，磁盘数据冗余存储，提升容错能力，确保数据安全。
- 无本地磁盘的天翼云物理机服务器，支持从云硬盘启动（即系统盘为云硬盘，且称之为快速发放物理机服务器）。云服务器备份可以对物理机服务器提供备份保护，支持基于多块云硬盘一致性快照技术的备份服务，并支持利用备份数据恢复服务器数据，最大限度保障用户数据的安全性和正确性，确保业务安全。

## 5.1.2 计费类

本章节主要介绍物理机的计费类的问题。

### 5.1.2.1 在哪里查询物理机的价格？

不同区域、不同规格的物理机服务器价格有所差异，您可以在[产品价格详情](#)查询价格

### 5.1.2.2 预付费和后付费是什么意思？如何选择？

**预付费：**一般为包年包月的购买形式，先购买再使用，系统根据用户选择的物理机类型和数量计算购买金额，对用户帐户金额进行扣除。

**后付费：**按资源的实际使用时长计费，适用于月结客户或线下合同客户。

### 5.1.2.3 物理机服务器退订时怎么扣费？

退订时如何扣费要根据资源状态、资源使用时长等条件而定，具体规则参见下表。

| 退订场景                   | 退订次数    | 收取手续费 | 收取已消费金额 | 返还代金券 | 返还优惠券 |
|------------------------|---------|-------|---------|-------|-------|
| 七天无理由<br>全额退订<br>(3次内) | 每账号每年3次 | 否     | 否       | 是     | 否     |

|            |      |   |   |   |   |
|------------|------|---|---|---|---|
| 非七天无理由退订   | 不限次数 | 是 | 是 | 否 | 否 |
| 创建失败的资源退订  | 不限次数 | 否 | 否 | 是 | 否 |
| 未生效的资源订购退订 | 不限次数 | 否 | 否 | 是 | 否 |
| 未生效的续费周期退订 | 不限次数 | 否 | 否 | 是 | 否 |

#### 5.1.2.4 怎么设置“包年/包月”资源的自动续费?

为防止资源到期被删除，用户可为长期使用的“包年/包月”产品开通自动续费。

##### 操作步骤、

1. 登录管理控制台，选择“产品 > 产品中心 > 产品续订”。
2. 选择物理机的类型 **BMS** 进行刷选
3. 选择需要续费的物理机设置自动续订



The screenshot displays the 'Product Renewal' (产品续订) page. At the top, there are navigation tabs for '全部时间' (All time), '7天内到期' (Expires within 7 days), '10天内到期' (Expires within 10 days), '到期时间>=10天' (Expires >=10 days), '未到期' (Not expired), and '已到期' (Expired). Below these are filters for '全部' (All) and '已开通自动续订' (Automatic renewal enabled). The main table lists resources with columns: '产品名称' (Product name), '资源ID/订单号' (Resource ID/Order number), '资源状态' (Resource status), '倒计时' (Countdown), '续订周期' (Renewal cycle), and '时间' (Time). A resource with ID '8' is selected, and a red box highlights the '设置自动续订' button in the bottom right corner.

### 5.1.2.5 物理机到期了，多久会释放资源？

物理机到期后会自动冻结，物理机关机，冻结周期一般为 15 天，冻结期过后如不续费会删除实例。

### 5.1.2.6 物理机冻结后，如何解冻？

当物理机资源到期而未续费时，自动进入冻结期，资源被冻结，您不能访问和使用该资源，例如无法下载物理机中的数据。及时续费后，资源将被解冻。

## 5.1.3 购买类

本章节主要介绍物理机的购买类的问题。

### 5.1.3.1 申请开通物理机需要多久？

通常在 30 分钟之内开通成功，快速发放物理机只需 5 分钟左右即可开通成功。

### 5.1.3.2 为什么申请物理机的任务失败，但是在服务器列表中显示申请成功？

#### 问题描述：

在弹性 IP 资源不足的情况下，用户通过管理控制台申请一台绑定弹性 IP 的物理机，此时物理机申请成功，但是绑定弹性 IP 失败。这种情况下，物理机申请状态栏显示任务失败，但物理机列表页面显示服务器申请成功。

#### 根因分析：

- 物理机列表中呈现的是申请成功的物理机列表和详情。
- 申请状态栏显示的是申请物理机任务的执行状态，包括申请物理机任务的各个子状态，例如申请物理机资源、绑定弹性 IP 等子任务，只有所有子任务全部成功时，任务才会成功，否则都是失败任务。

对于物理机资源申请成功，但是绑定弹性 IP 失败的情况而言，该任务处理失败。对于成功

申请的物理机资源，会短时间内出现在物理机列表中，待系统完成回退操作后，将会在列表自动消失。

## 5.1.4 管理类

本章节主要介绍物理机的管理类的问题。

### 5.1.4.1 物理机的使用限制是什么？

禁止修改网络相关的配置，否则可能导致无法连接物理机。不支持操作系统内核升级。

### 5.1.4.2 物理机与弹性云服务器的主要区别是什么？

弹性云服务器由多个租户共享物理资源，而物理机的资源归用户独享。对于关键类应用或性能要求较高的业务（如 Oracle RAC、大数据集群、企业中间件系统），并且要求安全可靠的运行环境，使用物理机更合适。

### 5.1.4.3 物理机与物理服务器有什么区别？

物理机，让物理服务器具有了自动发放、自动运维、VPC 互联、支撑对接共享存储等云的能力。可以像虚拟机一样灵活的发放和使用，同时又具备了极致的计算、存储、网络能力。

### 5.1.4.4 “删除”按钮是什么意思？

“删除”是指删除已申请的物理机，删除时可以选择同时删除弹性 IP 及绑定的云硬盘。如果不选择，则会保留，您需要单独删除操作。

## 5.1.5 登录类

本章节主要介绍物理机的登录类的问题。

### 5.1.5.1 物理机开机或者重启后，无法登录或云硬盘丢失，如何解决？

#### 问题描述

物理机开机或重启后，无法登录或云硬盘丢失。

### 可能原因

网络拥塞或者丢包导致物理机获取 IP 或者挂载数据卷（云硬盘）失败。

### 解决方案

重启物理机，如果多次重启无效，请联系客服。

### 5.1.5.2 远程登录物理机时，对浏览器版本有什么要求？

用户采用远程登录方式访问物理机时，使用的浏览器应满足下表。

支持的浏览器版本

| 浏览器               | 版本       |
|-------------------|----------|
| Google Chrome     | 31.0 及以上 |
| Mozilla FireFox   | 27.0 及以上 |
| Internet Explorer | 10.0 及以上 |

### 5.1.5.3 远程登录物理机时界面操作无响应，如何解决？

#### 问题描述

远程登录物理机时，按“Enter”后界面无任何响应。

#### 可能原因

物理机操作系统内部配置不允许通过远程访问。

#### 解决方案

登录物理机，进入操作系统进行相关设置，各操作系统的配置有所不同，以下仅提供部分操作系统配置示例，

### 步骤 1 修改配置文件。

1 对于 SUSE Linux Enterprise Server 12 SP2/SUSE Linux Enterprise Server 12 SP1/Ubuntu 16.04 Server/CentOS Linux 7.3/EulerOS 2.2 操作系统，使用 vi 编辑器打开 “/etc/default/grub”，在 “GRUB\_CMDLINE\_LINUX” 字段内容后添加 “console=tty0 console=ttyS0”，如下图所示。

```
If you change this file, run 'grub2-mkconfig -o /boot/grub2/grub.cfg' afterwar
ds to update
/boot/grub2/grub.cfg.
GRUB_DISTRIBUTOR=""
GRUB_DEFAULT=saved
GRUB_HIDDEN_TIMEOUT=0
GRUB_HIDDEN_TIMEOUT_QUIET=true
GRUB_TIMEOUT=8
GRUB_CMDLINE_LINUX_DEFAULT="resume=/dev/sda1 splash=silent quiet showopts crashk
ernel=99M,high crashkernel=72M,low"
kernel command line options for failsafe mode
GRUB_CMDLINE_LINUX_RECOVERY=single
GRUB_CMDLINE_LINUX="console=tty0 console=ttyS0"
Uncomment to enable BadRAM filtering, modify to suit your needs
This works with Linux (no patch required) and with any kernel that obtains
the memory map information from GRUB (GNU Mach, kernel of FreeBSD ...)
#GRUB_BADRAM=0x01234567,0xfefefefe,0x89abcdef,0xefefefef
Uncomment to disable graphical terminal (grub-pc only)
GRUB_TERMINAL=gfxterm
The resolution used on graphical terminal
note that you can use only modes which your graphic card supports via VBE
you can see them in real GRUB with the command 'vbeinfo'
GRUB_GFXMODE=auto
"grub" 40L, 2090C 15,46 Top
```

1 对于 Oracle Linux 7.3/Red Hat Enterprise Linux 7.3 操作系统，使用 vi 编辑器打开 “/etc/sysconfig/grub”，在 “GRUB\_CMDLINE\_LINUX” 字段内容后添加 “console=tty0 console=ttyS0”，如下图所示。

```
GRUB_TIMEOUT=5
GRUB_DISTRIBUTOR="$(sed 's, release .*$,g' /etc/system-release)"
GRUB_DEFAULT=saved
GRUB_DISABLE_SUBMENU=true
GRUB_TERMINAL_OUTPUT="console"
GRUB_CMDLINE_LINUX="crashkernel=auto vconsole.font=latarcyrheb-sun16 rd.lvm.lv
ol/swap rd.lvm.lv=ol/root vconsole.keymap=us rhgb quiet "console=tty0 console=ttyS0"
GRUB_DISABLE_RECOVERY="true"
```

### 步骤 2 刷新配置。

1 对于 SUSE Linux Enterprise Server 12 SP2/Oracle Linux 7.3/Red Hat Enterprise Linux 7.3/CentOS Linux 7.3/EulerOS 2.2 操作系统，执行以下命令刷新。

```
stty -F /dev/ttyS0 speed 115200
```

```
grub2-mkconfig -o /boot/grub2/grub.cfg
```

```
systemctl enable serial-getty@ttyS0
```

1 对于 Ubuntu 16.04 Server 操作系统，执行以下命令刷新。

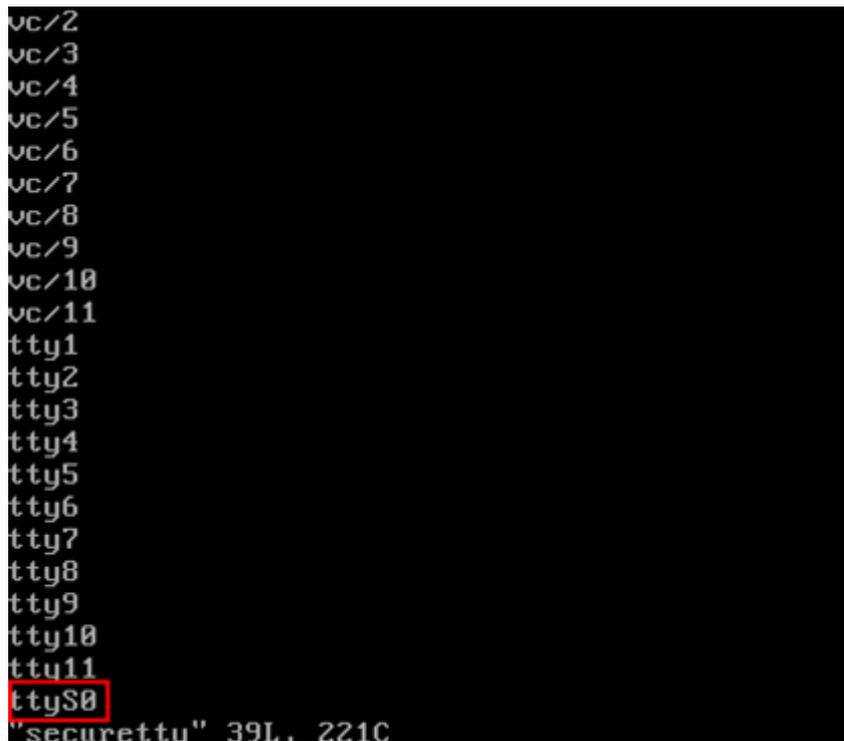
```
stty -F /dev/ttyS0 speed 115200
```

```
grub-mkconfig -o /boot/grub/grub.cfg
```

```
systemctl enable serial-getty@ttyS0
```

步骤 3 （可选）修改安全配置文件。

如果使用 root 用户通过串口进行登录，需要修改安全配置文件。在“/etc/securetty”最后添加如下信息：



```
vc/2
vc/3
vc/4
vc/5
vc/6
vc/7
vc/8
vc/9
vc/10
vc/11
tty1
tty2
tty3
tty4
tty5
tty6
tty7
tty8
tty9
tty10
tty11
ttyS0
"securetty" 39L, 221C
```

步骤 4 执行 reboot 重启操作系统。

———结束

物理机操作系统配置完成后，重新远程登录，确认是否可以登录成功。

### 5.1.5.4 远程登录物理机后控制台显示异常，如何解决？

#### 问题描述

远程登录物理机后，控制台出现如下异常现象：

使用 vim 编辑退出后，可编辑区域剩下半个屏幕。

输入字符长度超过 80 个时，当前行被覆盖。

使用 vim 等文本编辑器时，调整浏览器窗口大小后，屏幕出现错行，乱行等。

#### 可能原因

物理机远程登录受限于串口通信，无法自适应调整控制台屏幕大小，默认的控制台大小行列数为 24x80（24 为行数，80 为列数）。

#### 解决方案

远程登录物理机操作系统后，单击右键，选择“Resize: xxx”（如下图所示），此时命令行中会粘贴一条命令，比如 `stty cols 166 rows 48`，然后按“Enter”执行调整控制台大小操作。

```
Discovered PICMG Extension 2.2
Discovered IPMB-0 address 0x20
[SOL Session operational. Use ~? for help]
```

```
linux-8nad:~ # █
```



注意：

在使用 vim 或其它文本编辑器时，请勿调整浏览器窗口大小。如果需要改变浏览器窗口大小，请在退出 vim 等编辑器后，调整浏览器窗口大小，再按上述解决方案调整控制台大小。

## 5.1.6 网络与安全类

本章节主要介绍物理机的网络安全类的问题。

### 5.1.6.1 不同帐号下物理机内网是否可以互通？

不同帐号，即不同租户的物理机内网是不通的。

### 5.1.6.2 我申请的物理机是否在同一子网？

由于您可以自定义网络，所以无论是对于普通网络还是高速网络，物理机是否在一个子网，完全由您来控制。

### 5.1.6.3 我申请物理机可以和同 VPC 内的弹性云服务器通信吗？

可以。

您创建的 VPC，可能存在多个子网，物理机和弹性云服务器在同一个子网时进行二层通信，在不同网段时进行三层通信，不在同一个 vpc 内不能直接通信。

此外，物理机和弹性云服务器可以在同一个子网中，并且物理机与弹性云服务器通信时必须配置安全组规则。

如物理机有多个 vpc 网卡，则需要在物理机系统内配置策略路由。

### 5.1.6.4 在只能使用 SSH 登录物理机的情况下，如何修改物理机的网络配置或重启网络？

由物理机自动分配的网络是禁止修改的，在只有 SSH 登录的情况下修改，有可能会导物理机无法连接。如果物理机存在自定义 vlan 网络网卡，您可以配置或修改该网卡的网络。

### 5.1.6.5 物理机可以绑定多个弹性公网 IP 吗？

一个网卡只能绑定一个 EIP。您需要多个 EIP 时，可以将 EIP 绑定到扩展网卡，但扩展网卡绑定 EIP 以后，需要在物理机内根据实际网络情况做相应的操作，例如：增加策略路由或者命名空间等，来保证网络通信正常。

### 5.1.6.6 物理机可以绑定虚拟 ip 吗？

您可以绑定虚拟 IP 地址，为网卡提供额外的 IP 地址，从而实现更灵活的网络功能。

登录管理控制台。

单击管理控制台左上角的 ，选择地域和项目。

选择“计算 > 物理机”。

- 1、进入物理机页面。
- 2、单击待绑定虚拟 IP 地址的物理机的名称。系统跳转至该物理机的详情页面。
- 3、选择“网卡”页签，并单击“管理虚拟 IP 地址”。系统跳转至虚拟私有云页面。
- 4、在“虚拟 IP”页签，您可以在列表中选择合适的虚拟 IP，或者单击“申请虚拟 IP 地址”创建新的虚拟 IP 地址。
- 5、单击“操作”列下的“绑定服务器”，选择目标物理机与对应的网卡，绑定虚拟 IP 地址。

## 5.1.7 磁盘类

本章节主要介绍物理机的磁盘类的问题。

### 5.1.7.1 物理机是否支持挂载云硬盘？可以挂载多少块数据盘？

是，物理机支持挂载超高 I/O、高 I/O、普通 I/O 三种类型的云硬盘。

最多可以挂载 40 块数据盘。

说明

对于已有的物理机，如果需要进行挂载磁盘操作，磁盘的类型必须为“SCSI”才能挂载成功（目前磁盘类型支持 SCSI-3）。如果您需要新建云硬盘进行挂载，在创建页面上，云硬盘的设备类型必须勾选“SCSI”。

### 5.1.7.2 物理机挂载磁盘时有什么限制？

待挂载的磁盘与物理机属于同一可用分区。

物理机的状态为“运行中”或“关机”。

云硬盘的设备类型必须为“SCSI”。

如果是非共享盘，待挂载的云硬盘为“可用”状态。

如果是共享盘，待挂载的云硬盘为“正在使用”或“可用”状态。

### 5.1.7.3 如何查看云硬盘盘符？

物理机重启后，挂载至该物理机上的云硬盘盘符可能发生变化，您可以参考本指导快速找到云硬盘和盘符间的对应关系。

在物理机详情页面，记录云硬盘的“设备标识”信息。

在物理机详情页面，记录云硬盘的“设备标识”信息。

登录物理机操作系统，进入“/dev/disk/by-id”目录，查看 wwn 号和盘符的对应关系。其中 wwn 号在 Linux 操作系统中是由“wwn-0x + 设备标识”组成，例如：wwn-0x50000397c80b685d -> ../../sdc

```

rwxrwxrwx. 1 root root 9 Mar 20 17:20 wwn-0x50000397c8088c61 -> ../../sdb
rwxrwxrwx. 1 root root 9 Mar 20 17:20 wwn-0x50000397c80b2539 -> ../../sde
rwxrwxrwx. 1 root root 9 Mar 20 17:20 wwn-0x50000397c80b685d -> ../../sdc
rwxrwxrwx. 1 root root 9 Mar 20 17:20 wwn-0x50000397c80ba3e9 -> ../../sdg
rwxrwxrwx. 1 root root 9 Mar 20 17:20 wwn-0x50000397c80bb905 -> ../../sdf
rwxrwxrwx. 1 root root 9 Mar 20 17:20 wwn-0x50000397c810e531 -> ../../sdd
rwxrwxrwx. 1 root root 9 Mar 20 17:20 wwn-0x600508e000000002ab14603b88fa90b -> ../../sda
rwxrwxrwx. 1 root root 10 Mar 20 17:20 wwn-0x600508e000000002ab14603b88fa90b-part1 -> ../../sda1
rwxrwxrwx. 1 root root 10 Mar 20 17:20 wwn-0x600508e000000002ab14603b88fa90b-part2 -> ../../sda2
rwxrwxrwx. 1 root root 10 Mar 20 17:20 wwn-0x600508e000000002ab14603b88fa90b-part3 -> ../../sda3
rwxrwxrwx. 1 root root 10 Mar 20 17:20 wwn-0x600508e000000002ab14603b88fa90b-part4 -> ../../sda4
rwxrwxrwx. 1 root root 10 Mar 20 17:20 wwn-0x600508e000000002ab14603b88fa90b-part5 -> ../../sda5
rwxrwxrwx. 1 root root 9 Mar 20 17:20 wwn-0x68886030000369fafa17a17502223655 -> ../../sdh
rwxrwxrwx. 1 root root 10 Mar 20 17:20 wwn-0x68886030000369fafa17a17502223655-part1 -> ../../sdh1
rwxrwxrwx. 1 root root 10 Mar 20 17:20 wwn-0x68886030000369fafa17a17502223655-part2 -> ../../sdh2
rwxrwxrwx. 1 root root 10 Mar 20 17:20 wwn-0x68886030000369fafa17a17502223655-part3 -> ../../sdh3
rwxrwxrwx. 1 root root 9 Mar 21 14:16 wwn-0x6888603000036b61fa17a17502223655 -> ../../sdo

```

说明

建议租户在应用中使用 wwn 号进行磁盘操作，例如挂载磁盘：`mount wwn-0x50000397c80b685d 文件夹名称`。不建议直接使用盘符，因为会有盘符漂移而引起的无法找到磁盘的风险。

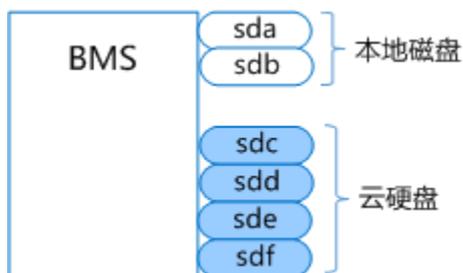
通过 wwn 号查找磁盘盘符，只支持 Linux 操作系统。

### 5.1.7.4 云硬盘设备名称与操作系统内块设备名称是否一致?

本地系统盘场景

物理机在详情页面显示的云硬盘设备名称与操作系统内部的设备名称不一致。为防止设备名称变化对业务造成影响，建议通过 UUID 的方式使用云硬盘，特别是/etc/fstab 的写入，最好使用 UUID，不然重启机器时可能导致无法正常启动，查看磁盘 UUID 命令为 blkid。

当携带云硬盘创建物理机完成后，物理机详情界面的云硬盘名称从/dev/sdb 开始进行显示，操作系统内部的云硬盘名称在本地硬盘名称之后，按照字母顺序依次增加，操作系统内部设备名称如下图所示。



物理机创建完成后进行挂载云硬盘，物理机详情界面的云硬盘名称为挂载云硬盘时指定的名称；卸载云硬盘后，详情界面将不再显示该云硬盘，同时释放该云硬盘对应的设备名称。

而物理机创建完成后进行卸载云硬盘，操作系统中云硬盘名称跟操作系统是否重启有关。

若操作系统为未重启，挂载云硬盘时操作系统会选择一个可用的、字母序中最小的盘符名称，比如/dev/sda 和/dev/sdc 已被使用，将会分配 dev/sdb；卸载云硬盘时操作系统会将云硬盘对应的设备名称释放。

若操作系统重启，操作系统内部的云硬盘名称会根据设备挂载时间以及本地磁盘个数重新生成，挂载云硬盘重启前后的现象，如图 1 所示；卸载云硬盘重启前后的现象如图 2 所示。

图 1 挂载云硬盘

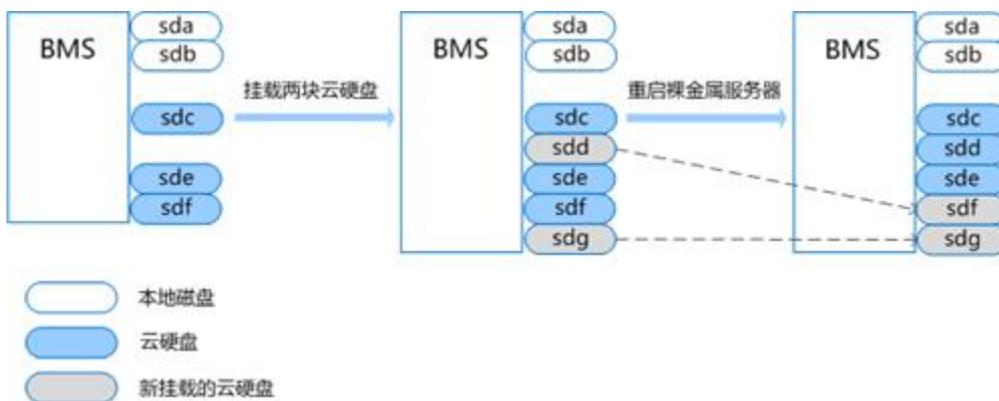
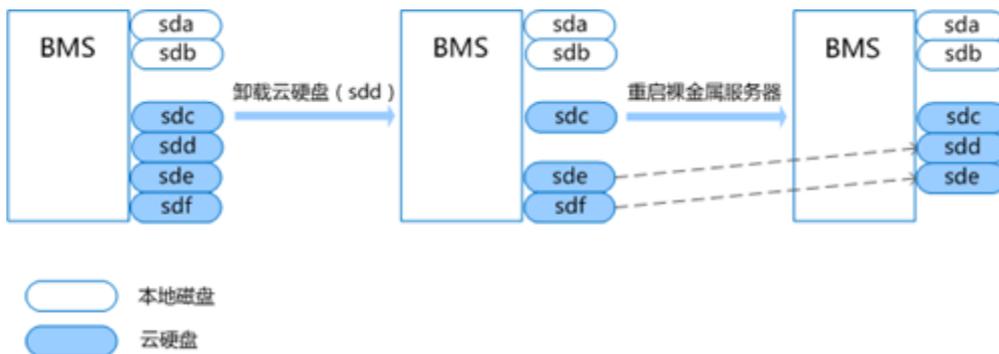


图 2 卸载云硬盘



### 云系统盘场景

物理机在详情页面显示的云硬盘设备名称与操作系统内部的设备名称可能不一致。

当携带云硬盘创建物理机完成后，物理机详情界面的云硬盘名称从/dev/sda 开始进行显示，操作系统内部的云硬盘名称和本地硬盘名称取决于系统的扫描顺序，整体按照字母顺序依次增加，可能存在以下两种情况，其中云系统盘始终为云硬盘盘符最小的一个，操作系统内部设备名称如图 3 和图 4 所示。

图 3 操作系统内部设备名称（一）

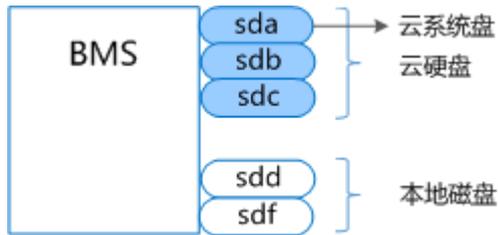
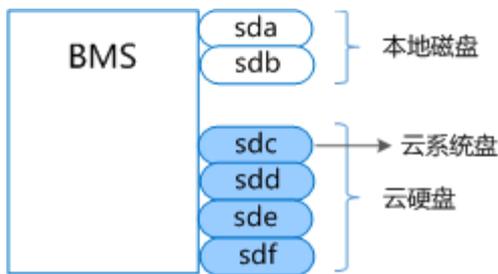


图 4 操作系统内部设备名称（二）



物理机创建完成后进行挂载云硬盘，物理机详情界面的云硬盘名称为挂载云硬盘时指定的名称；卸载云硬盘后，详情界面将不再显示该云硬盘，同时释放该云硬盘对应的设备名称。

而物理机创建完成后进行卸载云硬盘，操作系统中云硬盘名称跟操作系统是否重启有关。

若操作系统未重启，挂载云硬盘时操作系统会选择一个可用的、字母序中最小的盘符名称，比如/dev/sda 和/dev/sdc 已被使用，将会分配 dev/sdb；卸载云硬盘时操作系统会将云硬盘对应的设备名称释放。

若操作系统重启，操作系统内部的云硬盘名称会根据设备挂载时间以及本地磁盘个数重新生成，挂载云硬盘重启前后的现象，如图 5 和图 6 所示；卸载云硬盘重启前后的现象如图 7 和图 8 所示。

图 5 挂载云硬盘（重启前）

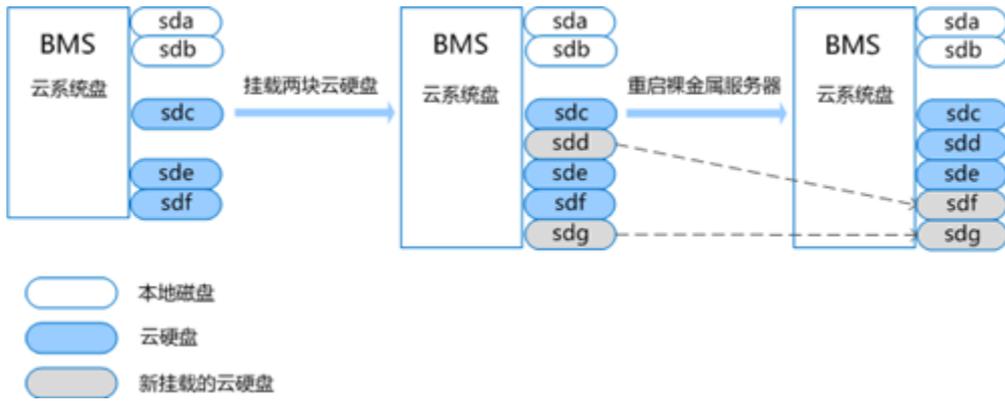


图 6 挂载云硬盘（重启后）

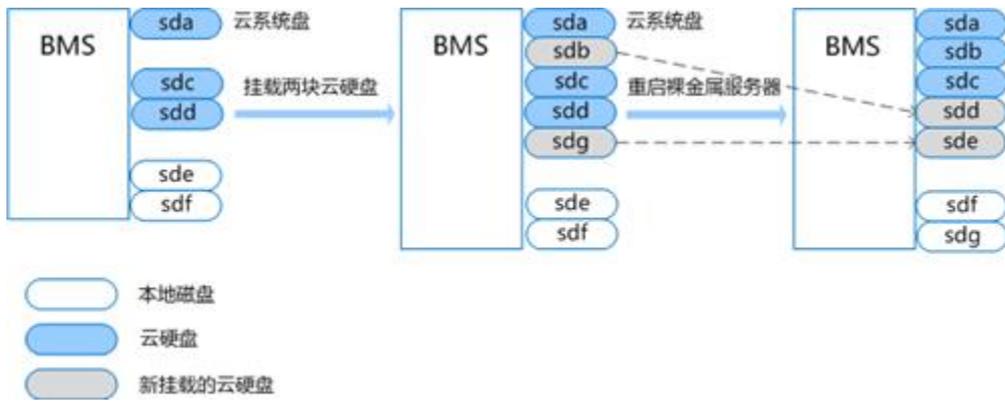


图 7 卸载云硬盘（重启前）

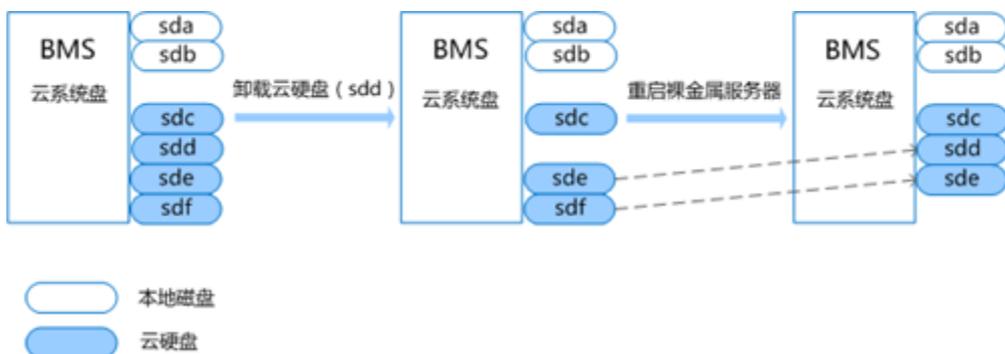
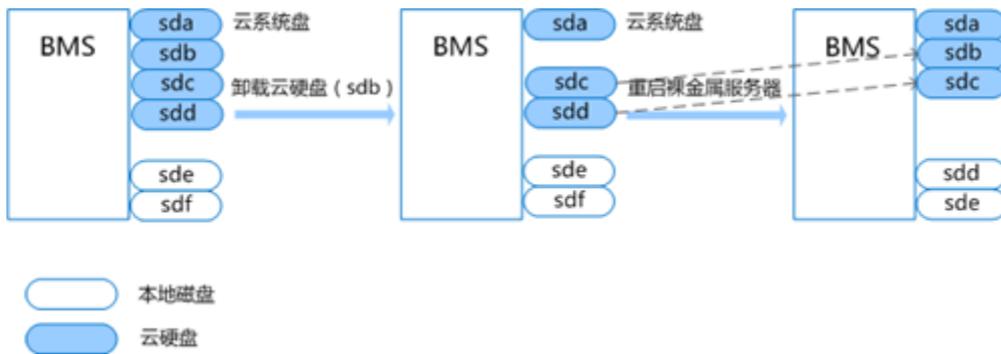


图 8 卸载云硬盘（重启后）



说明

登入物理机操作系统后，通过 `lsscsi` 命令，查看卷类型为“Huawei”的盘即为云硬盘。

### 5.1.7.5 为什么物理机操作系统内查看的容量大小比官网标称中容量要小？

在操作系统中查看硬盘容量时，获得的容量可能会比 BMS 服务官网标称的容量小，造成该差异的因素有以下几类：

硬件厂商对于存储容量的计算方式和操作系统不同。硬件厂商在生产硬盘时采用十进制来计算容量， $1\text{GB}=1000*1000*1000$  字节，而操作系统中以二进制计算容量， $1\text{GB}=1024*1024*1024$  字节，如 600G 在系统内显示就为 557.9G。

系统盘上的隐藏分区。如操作系统的引导分区、系统备份和还原分区等。

文件系统的开销。操作系统在使用硬盘前，需要先对硬盘分区及初始化文件系统，这些配置也会占用少量磁盘空间。

RAID 阵列占用的磁盘空间。如当有 2 块 600GB 的硬盘盘组 RAID 1 时，只能使用一块磁盘 600GB 的空间。

## 5.1.8 操作系统类

本章节主要介绍物理机的操作系统类的问题。

### 5.1.8.1 物理机能否更换操作系统？

暂时不支持更换操作系统。

### 5.1.8.2 物理机操作系统是否有图形界面？

当前提供的 Linux 操作系统是命令行界面，用户如果需要可以自己设置图形管理。或者订制专属物理机镜像。

### 5.1.8.3 物理机操作系统自带上传工具吗？

物理机操作系统没有自带的上传工具，需要用户自行安装和配置，例如 FTP。

### 5.1.8.4 如何设置物理机的静态主机名？

问题描述

Linux 物理机的静态主机名来源于创建物理机时，通过控制台注入的用户自定义名称。可以使用控制台或 `hostname` 命令更改物理机的主机名，但如果重启物理机，将自动还原为控制台上注入的用户自定义名称。

使用 `hostname` 命令修改 Linux 物理机的主机名后，如果您希望修改后的主机名在物理机关机、重启后仍然生效，需要将其写入相应的配置文件中，使之永久生效。

假设使用 `hostname` 命令修改后的主机名为 `new_hostname`。

步骤 1 修改配置文件“`/etc/hostname`”。

1. 执行以下命令，编辑配置文件“`/etc/hostname`”。

```
sudo vim /etc/hostname
```

2. 将配置文件中的主机名替换为修改后的主机名 `new_hostname`。

3. 执行以下命令，保存并退出文件。

```
:wq
```

步骤 2 对于 RHEL/CentOS/Fedora 6 操作系统，需要修改配置文件“/etc/sysconfig/network”。

1. 执行以下命令，编辑配置文件“/etc/sysconfig/network”。

```
sudo vim /etc/sysconfig/network
```

2. 修改涉及“HOSTNAME”的参数值，将其设置为修改后的主机名 new\_hostname。

```
HOSTNAME=new_hostname
```

3. 执行以下命令，保存并退出文件。

```
:wq
```

步骤 3 修改配置文件“/etc/cloud/cloud.cfg”。

1. 执行以下命令，编辑配置文件“/etc/cloud/cloud.cfg”。

```
sudo vim /etc/cloud/cloud.cfg
```

2. 按需选择如下两种方法，修改配置文件。

– 方法一：修改或添加“preserve\_hostname”参数项。

如果文件“/etc/cloud/cloud.cfg”中已有参数项“preserve\_hostname: false”，将其修改为“preserve\_hostname: true”即可。

如果文件“/etc/cloud/cloud.cfg”中没有该参数项，需要在“cloud\_init\_modules”模块前，添加语句 preserve\_hostname: true。

– 方法二：删除或者注释如下语句。

```
update_hostname
```

3. 执行以下命令，保存并退出文件。

```
:wq
```

步骤 4 修改物理机网络配置脚本“bms-network-config.conf”。

“bms-network-config.conf”的配置参数“enable\_preserve\_hostname”默认为 False，表示每次单板复位自动刷新主机名。将其修改为 True 即可关闭该功能。

1. 执行以下命令，编辑配置脚本“bms-network-config.conf”。

```
sudo vim /opt/huawei/network_config/bms-network-config.conf
```

2. 将参数项“enable\_preserve\_hostname”设置为 True。

```
enable_preserve_hostname: True
```

3. 执行以下命令，保存并退出文件。

```
:wq
```

步骤 5 对于 SUSE 系列操作系统，还需要修改配置文件“/etc/sysconfig/network/dhcp”。

1. 执行以下命令，编辑配置文件“/etc/sysconfig/network/dhcp”。

```
sudo vim /etc/sysconfig/network/dhcp
```

2. 将参数项“DHCLIENT\_SET\_HOSTNAME”设置为 no，保证每次 dhcp 服务不会自动分配主机名。

```
DHCLIENT_SET_HOSTNAME="no"
```

3. 执行以下命令，保存并退出文件。

```
:wq
```

步骤 6 执行以下命令，重启物理机。

```
sudo reboot
```

步骤 7 执行以下命令，验证静态主机名的修改是否永久生效。

```
sudo hostname
```

如果回显的内容是修改后的主机名 new\_hostname，表示主机名修改成功，永久生效。

### 5.1.8.5 如何设置物理机镜像密码的有效期？

如果物理机镜像密码已过期导致无法登录，请联系管理员处理。

如果物理机还可正常登录，用户可以参考以下操作设置密码有效期，避免密码过期造成的不便。

步骤 1 登录物理机操作系统，执行以下命令查看密码有效期。

```
vi /etc/login.defs
```

找到配置项“PASS\_MAX\_DAYS”，该参数表示密码的有效时间。

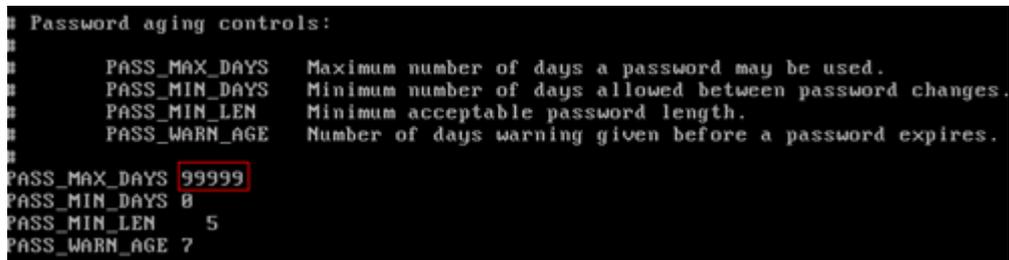
步骤 2 执行以下命令，修改步骤 1 中“PASS\_MAX\_DAYS”参数的取值。

```
chage -M 99999 user_name
```

其中，99999 为密码有效期限，user\_name 为系统用户。

建议用户根据实际情况及业务需求进行配置，定期使用该命令更新密码有效期。

步骤 3 再次执行 vi /etc/login.defs，验证配置是否生效。



```
Password aging controls:
#
PASS_MAX_DAYS Maximum number of days a password may be used.
PASS_MIN_DAYS Minimum number of days allowed between password changes.
PASS_MIN_LEN Minimum acceptable password length.
PASS_WARN_AGE Number of days warning given before a password expires.
#
PASS_MAX_DAYS 99999
PASS_MIN_DAYS 8
PASS_MIN_LEN 5
PASS_WARN_AGE 7
```

### 5.1.8.6 Windows 物理机的系统时间与本地时间相差 8 小时，如何处理？

问题原因

Linux 操作系统以主机板 CMOS 芯片的时间作为格林尼治标准时间，再根据设置的时区来确定系统的当前时间。但是一般 Windows 操作系统并不如此，Windows 系统直接将 CMOS 时间认定为系统当前时间，不再根据时区进行转换。

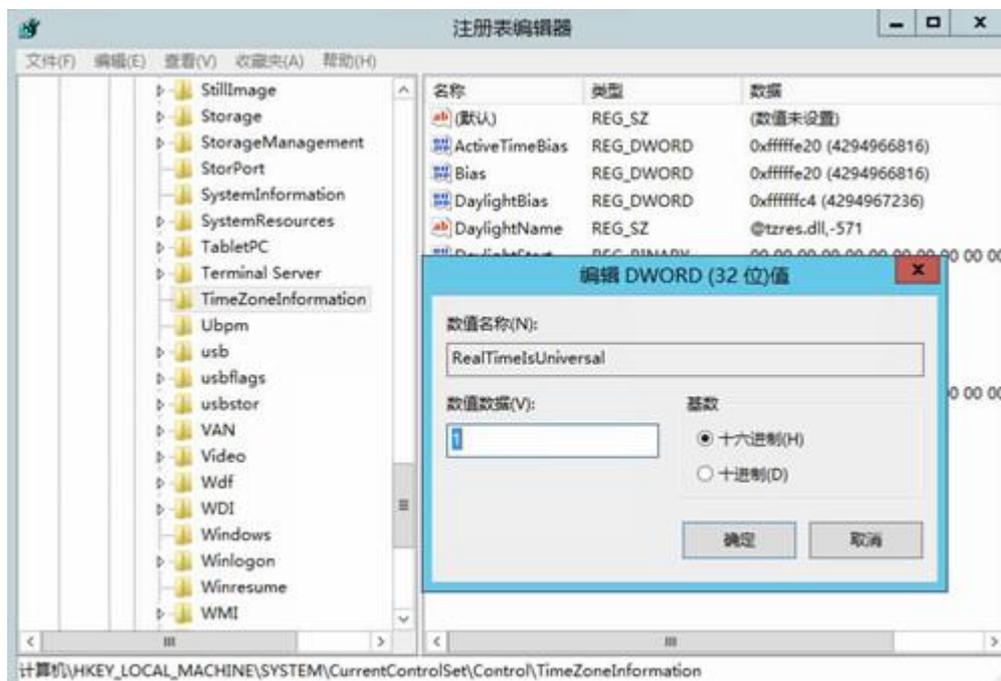
解决方案

步骤 1 登录 Windows 物理机操作系统。

步骤 2 单击“开始”菜单，在“搜索程序和文件”中输入“regedit”，并按“Enter”，打开注册表。

步骤 3 在“注册表编辑器”页面，选择“HKEY\_LOCAL\_MACHINE > SYSTEM > CurrentControlSet > Control > TimeZoneInformation”。

步骤 4 在“TimeZoneInformation”右侧区域右键单击，选择“新建 > DWORD (32-位)值(D)”，添加一项类型为 REG\_DWORD 的机码，名称为 RealTimeIsUniversal，值为 1。



步骤 5 修改完成后，重启物理机。

重启完成后，物理机时间和本地时间保持一致。

———结束

### 5.1.8.7 如何激活 Windows 物理机?

对于 Windows 系列操作系统，目前需要手动激活。

步骤 1 登录 Windows 物理机操作系统。

步骤 2 单击“开始”菜单，在“搜索程序和文件”中输入“cmd”，并按“Enter”，打开命令提示符。

步骤 3 执行以下命令，配置 KMS 服务器地址。

```
slmgr.vbs -skms 100.125.0.31
```

步骤 4 执行以下命令，查看是否激活。

```
slmgr.vbs -ato
```

如果出现错误：0xC004F074 软件授权服务器报告无法激活该物理机。密钥管理服务（KMS）不可用，说明无法激活，需要执行步骤 5。

步骤 5 查看物理机时间与标准时间是否一致，时间相差较大会出现无法激活的情况，将其设置为一致。

步骤 6 执行以下命令，检查物理机到 KMS 服务器端口是否可达。

```
telnet 100.125.0.31:1688
```

如果无法连接，说明物理机内部防火墙没有放通 1688 端口，需要关闭或者放通防火墙 TCP 1688 端口。如果有安全狗之类的安全软件也请暂时停止使用。

步骤 7 执行以下命令，重试物理机是否激活。

```
Slmgr.vbs -ato
```