

华为 TM210 灵活 IO 卡

用户指南

文档版本

03

发布日期

2020-05-25



华为技术有限公司



版权所有 © 华为技术有限公司 2020。保留一切权利。

非经本公司书面许可，任何单位和个人不得擅自摘抄、复制本文档内容的部分或全部，并不得以任何形式传播。

商标声明



HUAWEI和其他华为商标均为华为技术有限公司的商标。

本文档提及的其他所有商标或注册商标，由各自的所有人拥有。

注意

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

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

华为技术有限公司

地址： 深圳市龙岗区坂田华为总部办公楼 邮编： 518129

网址： <https://e.huawei.com>

前言

概述

本文档详细描述了TM210灵活IO卡的产品外观，功能以及基本配置方法。





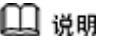
读者对象

本文档主要适用于以下工程师：

- 企业管理员
- 企业终端用户

符号约定

在本文中可能出现下列标志，它们所代表的含义如下。

符号	说明
 危险	表示如不避免则将会导致死亡或严重伤害的具有高等级风险的危害。
 警告	表示如不避免则可能导致死亡或严重伤害的具有中等级风险的危害。
 注意	表示如不避免则可能导致轻微或中度伤害的具有低等级风险的危害。
 须知	用于传递设备或环境安全警示信息。如不避免则可能会导致设备损坏、数据丢失、设备性能降低或其它不可预知的结果。 “须知”不涉及人身伤害。
 说明	对正文中重点信息的补充说明。 “说明”不是安全警示信息，不涉及人身、设备及环境伤害信息。

修改记录

文档版本	发布日期	修改说明
03	2020-05-25	第三次正式发布。 <ul style="list-style-type: none">• 2.1 特性列表增加支持特性：“虚拟机多队列”和“DPDK”。• 2.2 特性介绍增加“虚拟机多队列”和“DPDK”特性介绍。
02	2020-04-07	第二次正式发布。 修改 1.3 物理规格 中的“芯片型号/厂家”。
01	2019-12-14	第一次正式发布。

目录

前言.....	ii
1 产品简介.....	1
1.1 概述.....	1
1.2 外观.....	1
1.3 物理规格.....	2
2 产品特性.....	4
2.1 特性列表.....	4
2.2 特性介绍.....	5
3 兼容性.....	8
4 配置 TM210.....	9
4.1 TM210 的端口识别.....	9
4.2 驱动安装.....	10
4.2.1 获取驱动包.....	10
4.2.2 CentOS 7.6 下安装驱动.....	11
4.2.3 Ubuntu 18.04.1 下安装驱动.....	12
4.3 TM210 固件升级.....	14
4.4 Linux 下配置 VLAN.....	14
4.4.1 临时生效.....	14
4.4.2 永久生效.....	15
4.5 Linux 下配置端口 bonding.....	16
4.6 Linux 下配置端口 SR-IOV.....	17
4.6.1 使能 SR-IOV.....	17
4.6.2 创建虚拟机.....	18
4.6.3 添加 SR-IOV 网口.....	25
A 登录服务器实时桌面.....	28
B 缩略语.....	31

1 产品简介

1.1 概述

1.2 外观

1.3 物理规格

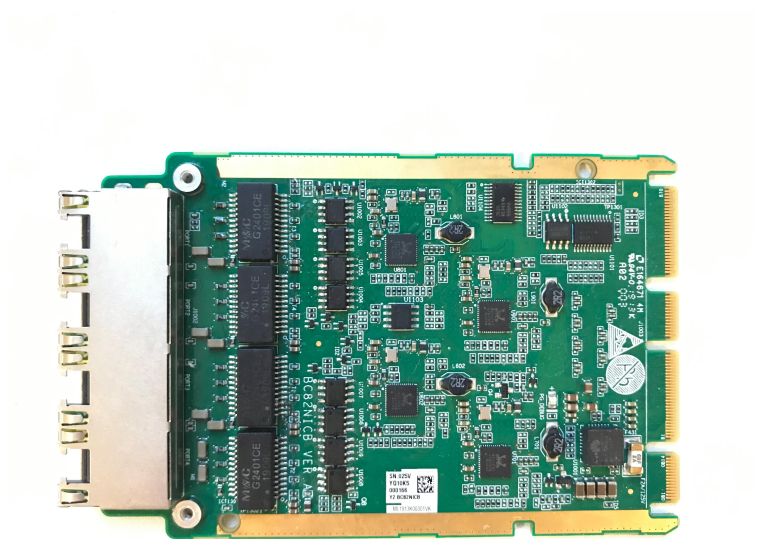
1.1 概述

TM210网卡是一款灵活IO卡，提供4个GE端口来支持NIC（Network Interface Card）应用。

1.2 外观

TM210网卡模块的外观图1-1所示。

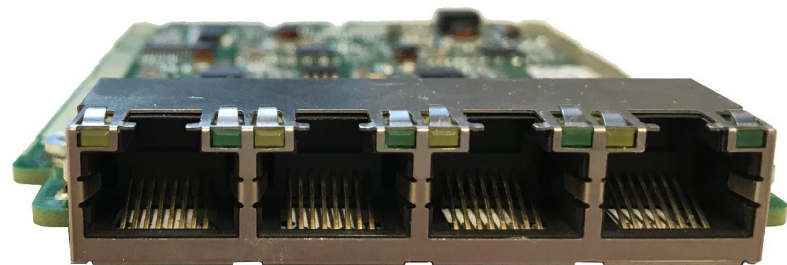
图 1-1 TM210 网卡



面板

TM210网卡面板如图1-2所示。

图 1-2 面板



说明

网口排序从左到右依次为1~4，其中每个网口左上角为ACT指示灯，右上角为Link指示灯。

指示灯

TM210网卡提供Link和ACT指示灯，各个指示灯具体状态如表1-1所示。

表 1-1 TM210 指示灯说明

指示灯类型	状态
Link指示灯	绿色常亮：链路有Link。
	绿色闪烁（2Hz）：该网口被定位。
	熄灭：链路断开。
ACT指示灯	黄色闪烁（1Hz）：有数据传输。
	常亮：网卡无数据传输。
	熄灭：链路断开。

1.3 物理规格

TM210的物理规格如表1-2所示。

表 1-2 TM210 的物理规格

参数类型	参数值
接口速率	1Gbps（电口），TM210网卡支持的工作模式及速率如表1-3所示。

参数类型	参数值
端口数量	4个
端口协议	Ethernet
端口形态	电口（RJ45）
芯片型号/厂家	CPU Integration/Huawei

表 1-3 TM210 工作模式及速率

速率协商模式	支持的速率	不支持的速率
自协商1000M Full	1000M	100M/10M
自协商100M Full/Half	100M	10M
自协商10M Full/Half	10M	-

2 产品特性

2.1 特性列表

2.2 特性介绍

2.1 特性列表

TM210支持以下的功能特性：

- 支持MAC VLAN表。TM210使用U/M（Unicast/Multicast）VLAN表进行MAC+VLAN或者MAC转发和过滤。
- 支持VLAN表。TM210为每个VLAN定义了一组成员，提供VLAN域内过滤、广播、洪泛等功能。
- 支持流表，通过流表对报文进行过滤和转发。
- 支持流控，有效防止报文丢失。
- 支持MTU，支持9.5kB的Jumbo帧。
- 支持网卡混杂模式。
- 支持GRO（Generic receive offload），把小包封装成大包再递交给协议栈。
- 支持VLAN虚拟局域网。
- 支持VxLAN，创建第2层逻辑网络，并将其封装在标准第3层IP数据包中。
- 支持TSO（TCP Segment Offload）。
- 支持RSS（receive side scaling）。
- 支持网卡队列设置。
- 支持网卡Checksum功能。
- 支持网卡点灯。
- 支持网卡SR-IOV（Single Root I/O Virtualization）。
- 支持虚拟机多队列（Virtual Machine Multi Queue）。
- 支持DPDK（Data Plane Development Kit）。
- 支持网卡PXE功能。
- 支持IPv4/IPv6。

2.2 特性介绍

MAC VLAN 表

TM210使用U/M（Unicast/Multicast）VLAN表进行MAC+VLAN或者MAC转发和过滤。总共支持4K+128条表项，其中最大支持4K+128条单播表项，1K组播表项。

VLAN 表

VLAN表为每个VLAN定义了一组成员，提供VLAN域内过滤、广播、洪泛等功能。VLAN过滤默认是不使能的，所有的报文将不会被VLAN过滤掉。

流表

流表是PPP的一部分，主要用于对报文进行过滤和转发。基本原理是先通过用户配置或自学习的方式创建流表表项，表项形式为key-action形式，key为匹配规则，包含报文的多种形式的tuple信息，action为匹配之后的对报文的操作，可以为丢弃、转发到指定队列、转发到其他表进一步查表。

流控

流控通过实时了解对端设备对通信流量的要求，根据对端设备的要求来决定本设备是继续发送还是暂停发送数据包。

MTU

MTU即Maximum Transmission Unit（最大传输单元），是指一种通信协议的某一层上所能通过的最大数据包大小。通常人们所说的MTU为IP层的最大传输数据大小。TM210支持9.5kB的Jumbo帧。

混杂模式

混杂模式是指接收所有经过网卡的数据包，包括不是发给本机的包，即不验证MAC地址。网卡的缺省工作模式包含广播模式和直接模式，即它只接收广播帧和发给自己的帧。

GRO

GRO是在协议栈接收报文时进行减负的一种处理方式，该方式在设计上考虑了多种协议报文。GRO功能在接收端通过把多个相关的报文（比如TCP分段报文）组装成一个大的报文后再传送给协议栈进行处理，因此减少了协议栈处理报文个数，加快协议栈对报文的处理速度。

VLAN

VLAN（Virtual Local Area Network）即虚拟局域网，是将一个物理的LAN在逻辑上划分成多个广播域的通信技术。TM210最多支持4095个VLAN ID。

VxLAN

VXLAN是一种网络虚拟化技术，创建第2层逻辑网络，并将其封装在标准第3层IP数据包中。可以对二层网络在三层范围进行扩展，应用于数据中心，使虚拟机可以在互相连通的三层网络范围内迁移，而不需要改变IP地址和MAC地址。

TSO

TSO是一种利用网卡的少量处理能力，降低CPU发送数据包负载的技术，需要网卡硬件及驱动的支持。

RSS

RSS通过一定的hash算法实现接收方向上的负载均衡，使报文尽量平均地分配到不同队列，从而对应到不同的CPU核，降低因CPU切换调度带来的性能损失。

队列

在NIC领域，队列通常以QP（Queue Pair）方式存在，单个QP包含1个TX队列和1个RX队列。每个QP映射到一块存储空间，PF/VF通过申请1个或多个QP来实现配置和服务。当前TM210只支持一个队列，不可修改。

Checksum

校验功能，可以替代系统的TCP/IP协议栈来实现部分报文类型的校验和计算，如IP、TCP、UDP和SCTP。

点灯

点灯特指对网口设备的点灯，其LED指示灯可用于标识不同的工作状态，例Link状态，电源状态，故障状态等，网口点灯功能需要软硬件配合实现。

SR-IOV

TM210的GE端口支持SR-IOV功能。每个物理端口使能出来的VF可以添加给VM（Virtual Machine）使用，与VM建立映射关系。

VMMQ

VMMQ是一种NIC特性，它允许VM的流量分散到多个队列，每个队列由不同的物理处理器处理。然后，流量被传递到VM中的多个LP，就像它在vRSS中一样，这允许向VM提供大量的网络带宽。

DPDK

DPDK是一个用来进行包数据处理加速的软件库，专注于网络应用中数据包的高性能处理。

技术优点

- 通过UIO技术将报文拷贝到应用空间处理，规避不必要的内存拷贝和系统调用，便于快速迭代优化。
- 通过大页内存HUGEPAGE，降低cache miss（访存开销），利用内存多通道交错访问提高内存访问有效带宽，即提高命中率，进而提高cpu访问速度。

- 通过CPU亲和性，绑定网卡和线程到固定的core，减少cpu任务切换。特定任务可以被指定只在某个核上工作，避免线程在不同核间频繁切换，保证更多的cache命中。
- 通过无锁队列，减少资源竞争。cache行对齐，预取数据，多元数据批量操作。
- 通过轮询可在包处理时避免中断上下文切换的开销。

PXE

TM210的GE端口支持PXE。PXE用于通过Ethernet或IP网络进行远程启动，远端连接PXE Server提供OS加载。

3 兼容性

TM210支持的操作系统以及硬件的详细信息，请参见[智能计算产品兼容性查询助手](#)或咨询华为当地销售代表。

4 配置 TM210

- [4.1 TM210的端口识别](#)
- [4.2 驱动安装](#)
- [4.3 TM210固件升级](#)
- [4.4 Linux下配置VLAN](#)
- [4.5 Linux下配置端口bonding](#)
- [4.6 Linux下配置端口SR-IOV](#)

4.1 TM210 的端口识别

TM210的端口在OS下识别，下面使用CentOS 7.6系统为例进行说明。

步骤1 以root用户登录服务器OS，单击鼠标右键，选择“Open Terminal”打开命令行。

步骤2 执行如下命令，查看TM210的网卡设备（PCIe function）。

```
lspci | grep -i eth
```

回显如下：

```
7d:00.0 Ethernet controller: Huawei Technologies Co., Ltd. HNS GE/10GE/25GE RDMA Network Controller (rev 21)
7d:00.1 Ethernet controller: Huawei Technologies Co., Ltd. HNS GE/10GE/25GE Network Controller (rev 21)
7d:00.2 Ethernet controller: Huawei Technologies Co., Ltd. HNS GE/10GE/25GE RDMA Network Controller (rev 21)
7d:00.3 Ethernet controller: Huawei Technologies Co., Ltd. HNS GE/10GE/25GE Network Controller (rev 21)
```

步骤3 执行如下命令，查询到TM210的4个NIC端口。

```
ifconfig -a
```

TM210的4个NIC端口以“**enp125s0f[num]/enp189s0f[num]/eth[num]**”的形式显示，如回显中的“enp125s0f0”、“enp125s0f1”、“enp125s0f2”和“enp125s0f3”。具体端口名称与服务器的操作系统有关，操作系统不同，名称显示不同。

```
enp125s0f0: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
    ether 00:01:08:00:0e:58 txqueuelen 1000 (Ethernet)
    RX packets 0 bytes 0 (0.0 B)
    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

enp125s0f1: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
inet 128.5.251.163 netmask 255.255.255.0 broadcast 128.5.251.255
inet6 fe80::502e:dcf:71:cad4 prefixlen 64 scopeid 0x20<link>
ether 00:01:08:00:13:52 txqueuelen 1000 (Ethernet)
RX packets 48529 bytes 4060749 (3.8 MiB)
RX errors 0 dropped 6676 overruns 0 frame 0
TX packets 185 bytes 81044 (79.1 KiB)
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

enp125s0f2: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
ether 00:01:08:00:13:01 txqueuelen 1000 (Ethernet)
RX packets 0 bytes 0 (0.0 B)
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

enp125s0f3: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
inet 10.10.10.10 netmask 255.255.255.0 broadcast 10.10.10.255
inet6 fe80::7cd9:3aeb:beb8:4fb8 prefixlen 64 scopeid 0x20<link>
ether 00:01:08:00:10:34 txqueuelen 1000 (Ethernet)
RX packets 0 bytes 0 (0.0 B)
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
```

步骤4 执行如下命令，查看端口信息。

```
ethtool -i enp125s0f[num]/enp189s0f[num]/eth[num]
```

如查看enp125s0f0的端口信息，命令为：**ethtool -i enp125s0f0**

回显如下：

```
driver: hns3
version: 4.19.36-vhulk1906.3.0.h356.eule
firmware-version: 1.8.12.3
expansion-rom-version:
bus-info: 0000:7d:00.0
supports-statistics: yes
supports-test: yes
supports-eeprom-access: no
supports-register-dump: yes
supports-priv-flags: no
```

----结束

4.2 驱动安装

TM210网卡驱动默认使用操作系统自带的驱动，对于CentOS 7.6和Ubuntu 18.04.1系统，需要手动进行驱动安装。

下面分别介绍CentOS 7.6和Ubuntu 18.04.1系统下驱动安装的步骤。

4.2.1 获取驱动包

步骤1 登录[华为企业业务网站](#)。

步骤2 在导航栏中选择“技术支持 > 智能管理软件 > iDriver > TaiShanServer iDriver”，进入目标服务器的详细页面。

步骤3 选择“软件”页签。

步骤4 下载软件压缩包。

1. 单击目标版本，进入软件包列表页面。
2. 在“版本资料”列表中，选择TaiShan的驱动配套表并下载。
3. 版本配套表按照不同操作系统分为很多Sheet。目前只有CentOS 7.6和Ubuntu 18.04.1系统需要安装板载网卡驱动，选择“CentOS”或“Ubuntu”Sheet，通过“Card Name”过滤含有关键字“TM210”的行，查看“System Version”、“Driver File”和“Onboard ISO Driver contain Files”。

Verify the upgrade package by using the digital certificate.			Software digital signature (OpenPGP) validation tool		
External Driver Version	System Version	Driver File	Onboard ISO Driver contain Files	Card Name	Driver Version
CentOS-Driver	CentOS 7.6	onboard_driver_CentOS7.6.iso	NIC-hisi_eth-CentOS7.6-hns3-1.0.1-1-aarch64.rpm	TM210	1.0.1

说明

- System Version：操作系统版本。
- Driver File：软件压缩包名称。
- Onboard ISO Driver contain Files：板载ISO驱动包含文件，即所需的软件文件。

CentOS7.6所对应的System Version为“CentOS 7.6”，Driver File为“onboard_driver_CentOS7.6.iso”，驱动包文件为“NIC-hisi_eth-CentOS7.6-hns3-1.0.1-1-aarch64.rpm”；Ubuntu 18.04.1对应的System Version为“Ubuntu 18.04.1 LTS”，Driver File为“onboard_driver_Ubuntu18.04.1.iso”，驱动包文件为“NIC-hisi_eth-Ubuntu18.04.1-hns3-1.0.2-1-aarch64.deb”。

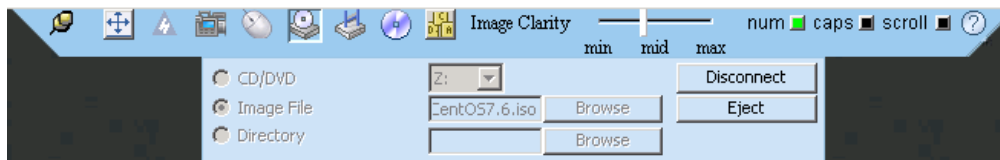
---结束

4.2.2 CentOS 7.6 下安装驱动

步骤1 参见[4.2.1 获取驱动包](#)下载驱动压缩文件，CentOS 7.6系统对应的压缩文件为“TaiShan iDriver-CentOS7.6-Driver-V100.zip”，将文件解压得到“onboard_driver_CentOS7.6.iso”镜像文件。

步骤2 打开iBMC远程虚拟控制台，具体操作步骤请参考[A 登录服务器实时桌面](#)。

步骤3 在iBMC远程虚拟控制台菜单点击光驱图标，选择“Image File”，单击“Browse”，选择已下载的驱动镜像文件“onboard_driver_CentOS7.6.iso”，单击“Connect”挂载驱动镜像文件。



步骤4 以root用户登录服务器OS，在桌面单击鼠标右键，选择“Open Terminal”打开命令行。

步骤5 执行如下命令，将驱动镜像文件挂载到“/mnt”路径下。

```
mount /dev/sr0 /mnt
```


说明

如服务器存在多个光驱设备，则虚拟光驱可能为sr1、sr2等名称，需依次进行挂载，直到确认/mnt目录下的文件为驱动文件。

步骤6 执行如下命令新建路径用于保存驱动文件，例如“/root/driver”。

```
mkdir /root/driver
```

步骤7 执行如下命令将挂载ISO文件后得到的驱动文件拷贝到系统下指定目录，例如“/root/driver”。

```
cp -vr /mnt/NIC-hisi_eth-CentOS7.6-hns3-1.0.1-1-aarch64.rpm /root/driver
```

步骤8 执行如下命令编辑dist-blacklist.conf文件。

```
vim /usr/lib/modprobe.d/dist-blacklist.conf
```

步骤9 按Insert键进入编辑模式，在dist-blacklist.conf文件末尾添加如下两行，按ESC键退出编辑模式，输入:wq!，按Enter键保存退出。

```
blacklist hns_roce_hw_v2
```

```
blacklist hns_roce
```

步骤10 执行如下命令重新生成initramfs文件。

```
mv /boot/initramfs-$(uname -r).img /boot/initramfs-$(uname -r).img.bak
```

```
dracut /boot/initramfs-$(uname -r).img $(uname -r)
```

步骤11 输入如下命令重启系统。

```
reboot
```

步骤12 系统启动后以root用户登录服务器OS，在桌面单击鼠标右键，选择“Open Terminal”打开命令行。

步骤13 执行如下命令切换到驱动文件所在目录。

```
cd /root/driver
```

步骤14 执行如下命令进行驱动安装。

```
rpm -ivh NIC-hisi_eth-CentOS7.6-hns3-1.0.1-1-aarch64.rpm
```

步骤15 执行如下命令查看驱动是否升级到驱动配套表中的指定版本。

```
modinfo hns3
```

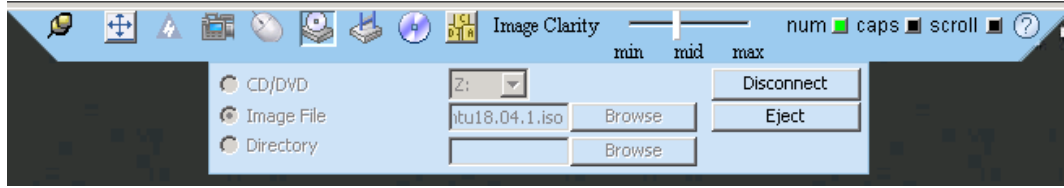
----结束

4.2.3 Ubuntu 18.04.1 下安装驱动

步骤1 参见[4.2.1 获取驱动包](#)获取驱动压缩文件，Ubuntu 18.04.1系统对应的压缩文件为“TaiShan iDriver-Ubuntu18.04.1-Driver-V100.zip”，将压缩文件解压得到“onboard_driver_Ubuntu18.04.1.iso”镜像文件。

步骤2 打开iBMC远程虚拟控制台，具体操作步骤请参考[A 登录服务器实时桌面](#)。

步骤3 在iBMC远程虚拟控制台菜单单击光驱图标，选择“Image File”，单击“Browse”，选择已下载的驱动镜像文件“onboard_driver_Ubuntu18.04.1.iso”，单击“Connect”挂载驱动镜像文件。



步骤4 以root用户登录服务器OS，在桌面单击鼠标右键，选择“Open Terminal”打开命令行。

步骤5 执行如下命令，将驱动镜像文件挂载到“/mnt”路径下。

```
mount /dev/sr0 /mnt
```

说明

如服务器存在多个光驱设备，则虚拟光驱可能为sr1、sr2等名称，需依次进行挂载，直到确认/mnt目录下的文件为驱动文件。

步骤6 执行如下命令新建路径用于保存驱动文件，例如“/root/driver”。

```
mkdir /root/driver
```

步骤7 执行如下命令将挂载ISO文件后得到的驱动文件拷贝到系统下指定目录，例如“/root/driver”。

```
cp -vr /mnt/NIC-hisi_eth-Ubuntu18.04.1-hns3-1.0.2-1-aarch64.deb /root/driver
```

步骤8 执行如下命令编辑blacklist.conf文件。

```
vim /etc/modprobe.d/blacklist.conf
```

步骤9 按Insert键进入编辑模式，在dist-blacklist.conf文件末尾添加如下两行，按"Esc"键退出编辑模式，输入":wq!"，按"Enter"键保存退出。

```
blacklist hns_roce_hw_v2
```

```
blacklist hns_roce
```

步骤10 执行如下命令重新生成initramfs文件。

```
update-initramfs -u
```

步骤11 输入如下命令重启系统。

```
reboot
```

步骤12 系统启动后以root用户登录服务器OS，在桌面单击鼠标右键，选择“Open Terminal”打开命令行。

步骤13 执行如下命令切换到驱动文件所在目录。

```
cd /root/driver
```

步骤14 执行如下命令进行驱动安装。

```
dpkg -i NIC-hisi_eth-Ubuntu18.04.1-hns3-1.0.2-aarch64.deb
```

步骤15 执行如下命令查看驱动是否升级到驱动配套表中的指定版本。

```
modinfo hns3
```

----结束

4.3 TM210 固件升级

TM210暂不支持固件升级。

4.4 Linux 下配置 VLAN

TM210支持tag和untag的报文通过，可在OS下配置VLAN ID，或者保持untag。

Linux下配置VLAN，有两种配置生效的方式：临时生效和永久生效。

- 重启网络服务影响其他业务的正常运行时，可采用[4.4.1 临时生效](#)。
- 重启网络服务不影响其他业务的正常运行时，可采用[4.4.2 永久生效](#)。

下面分别以CentOS 7.6和Ubuntu 18.04.1系统为例介绍在Linux下配置VLAN的操作步骤。

4.4.1 临时生效

步骤1 以root用户登录服务器OS，在桌面单击鼠标右键，选择“Open Terminal”打开命令行。

步骤2 执行如下命令找到NIC端口，以enp125s0f3网口为例。

```
ifconfig -a
```

步骤3 执行如下命令添加VLAN到端口，例如，为enp125s0f3添加VLAN 100。

```
ip link add link enp125s0f3 name enp125s0f3.100 type vlan id 100
```

步骤4 执行如下命令配置VLAN的IP地址，相当于网口子端口的IP地址，例如，为enp125s0f3端口100配置IP地址192.168.13.200/24。

```
ip addr add 192.168.13.200/24 dev enp125s0f3.100
```

步骤5 执行如下命令启用网卡。

```
ip link set dev enp125s0f3.100 up
```

步骤6 执行如下命令查询配置结果。

```
ifconfig enp125s0f3.100
```

回显如下信息：

```
enp125s0f3.100: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
    inet 192.168.13.200 netmask 255.255.255.0 broadcast 0.0.0.0
    ether 00:04:45:77:0c:b6 txqueuelen 1000 (Ethernet)
    RX packets 0 bytes 0 (0.0 B)
    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
```

说明

- 不同VLAN的IP地址要配置在不同的网段。
- 删除VLAN使用**ip link delete [Device Name].[VLAN ID]**命令，其中*Device Name*表示网卡设备名称（如enp125s0f3），*VLAN ID*表示VLAN ID编号（如100）。

----结束

4.4.2 永久生效

步骤1 以**root**用户登录服务器OS，在桌面单击鼠标右键，选择“Open Terminal”打开命令行。

步骤2 执行如下命令找到NIC端口，以enp125s0f3网口为例。

```
ifconfig -a
```

步骤3 执行如下进入网络配置文件目录。

```
cd /etc/sysconfig/network-scripts
```

步骤4 执行如下命令复制一个“ifcfg-enp125s0f3.100”的文件。

```
cp ifcfg-enp125s0f3 ifcfg-enp125s0f3.100
```

步骤5 执行如下命令修改“ifcfg-enp125s0f3.100”的文件中的配置。

```
vi ifcfg-enp125s0f3.100
```

按“i”修改以下备注说明的内容。

```
TYPE=Ethernet
PROXY_METHOD=none
BROWSER_ONLY=no
BOOTPROTO=static \\配置为静态IP地址
DEFROUTE=yes
IPV4_FAILURE_FATAL=no
IPV6INIT=yes
IPV6_AUTOCONF=yes
IPV6_DEFROUTE=yes
IPV6_FAILURE_FATAL=no
IPV6_ADDR_GEN_MODE=stable-privacy
NAME=eno2.100 \\配置为eno2.[vlan ID]
UUID=38c678e5-225d-4cc8-a7a4-db1b59ac98ef
DEVICE=eno2.100 \\配置为eno2.[vlan ID]
ONBOOT=yes \\配置为yes
IPADDR=192.168.13.100 \\配置IP地址
GATEWAY=192.168.13.1 \\配置网关
NETMASK=255.255.255.0 \\配置掩码
VLAN=yes \\配置为yes
```

编辑完成后按“Esc”退出编辑模式，输入“:wq!”保存并退出。

步骤6 执行如下命令重启网络服务使配置永久生效。

```
systemctl restart network.service
```

步骤7 执行如下查询配置结果。

```
ifconfig enp125s0f3.100
```

```
enp125s0f3.100: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
    inet 192.168.13.100 netmask 255.255.255.0 broadcast 192.168.13.255
    ether 00:04:45:77:0c:b6 txqueuelen 1000 (Ethernet)
    RX packets 0 bytes 0 (0.0 B)
```

```
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
```

说明

- 不同VLAN的IP地址要配置在不同的网段（subnet）。
- 删除VLAN时首先需要删除配置文件“ifcfg-enp125s0f3.100”，然后执行**systemctl restart network.service**命令重启网络服务。

----结束

4.5 Linux 下配置端口 bonding

下面以CentOS 7.6为例介绍端口bonding mode1配置。

步骤1 以root用户登录服务器OS，在桌面单击鼠标右键，选择“Open Terminal”打开命令行。

步骤2 执行如下命令找到NIC端口。

```
ifconfig -a
```

步骤3 执行如下命令进入“/etc/sysconfig/network-scripts”目录。

```
cd /etc/sysconfig/network-scripts
```

步骤4 创建并打开“ifcfg-bond0”。

```
vi ifcfg-bond0
```

按“i”编辑文件，在“ifcfg-bond0”文件中添加以下内容。

```
BOOTPROTO=none
NAME=bond1
TYPE=Bond
ONBOOT=yes
IPADDR=192.168.5.122 //填写IP地址，作为bonding端口的IP地址
NETMASK=255.255.255.0 //填写子掩码
BONDING_MASTER=yes
DEVICE=bond0
BONDING_OPTS="mode=1 miimon=100" //填写模式及链路检测时间
```

编辑完成后按“Esc”退出编辑模式，输入“:wq!”保存并退出。

步骤5 将“ifcfg-enp125s0f[num]”文件移动到“backup”目录进行备份。

```
mv /etc/sysconfig/network-scripts/ifcfg-eno3 /backup/ifcfg-enp125s0f2
```

```
mv /etc/sysconfig/network-scripts/ifcfg-eno4 /backup/ifcfg-enp125s0f3
```

步骤6 创建并打开“ifcfg-enp125s0f2”文件。

```
vi /backup/ifcfg-enp125s0f2
```

按“i”编辑文件，在“ifcfg-enp125s0f2”中添加以下内容。

```
DEVICE=enp125s0f2
BOOTPROTO=none
ONBOOT=yes
MASTER=bond0
SLAVE=yes
TYPE=Ethernet
```

编辑完成后按“Esc”退出编辑模式，输入“:wq!”保存并退出。

步骤7 创建并打开“ifcfg-enp125s0f3”文件。

vi /backup/ifcfg-enp125s0f3

按“i”编辑文件，在“ifcfg-enp125s0f3”中添加以下内容。

```
DEVICE=enp125s0f3
BOOTPROTO=none
ONBOOT=yes
MASTER=bond0
SLAVE=yes
TYPE=Ethernet
```

编辑完成后按“Esc”退出编辑模式，输入“:wq!”保存并退出。

步骤8 执行如下命令加载驱动模块。

modprobe bonding

步骤9 执行如下命令重启网络应用。

systemctl restart network.service

----结束

4.6 Linux 下配置端口 SR-IOV

4.6.1 使能 SR-IOV

下面以Ubuntu 18.04.31为例介绍在Linux下使能SR-IOV的操作步骤。

步骤1 以root用户登录服务器OS，在桌面单击鼠标右键，选择“Open Terminal”打开命令行。

步骤2 执行如下命令找到NIC的端口，以eth0为例。

ifconfig -a

步骤3 执行如下命令查看NIC端口支持的最大虚拟化网卡数量，命令执行结果会显示支持虚拟网口的最大数量。

cat /sys/class/net/enp125s0f3/device/sriov_totalvfs

步骤4 根据实际需求创建不大于端口最大虚拟化数量的虚拟化网口，以创建2个虚拟化网口为例。

echo 2 > /sys/class/net/enp125s0f3/device/sriov_numvfs

步骤5 执行如下命令查看新增VF对应的PCIe设备，其中7d:01.0和7d:01.1设备即为新增的VF。

lspci|grep -i eth

回显如下：

```
7d:00.0 Ethernet controller: Huawei Technologies Co., Ltd. Device a222 (rev 21)
7d:00.1 Ethernet controller: Huawei Technologies Co., Ltd. Device a221 (rev 21)
7d:00.2 Ethernet controller: Huawei Technologies Co., Ltd. Device a222 (rev 21)
7d:00.3 Ethernet controller: Huawei Technologies Co., Ltd. Device a221 (rev 21)
```

```
7d:01:0 Ethernet controller: Huawei Technologies Co., Ltd. Device a22f (rev 21)
7d:01:1 Ethernet controller: Huawei Technologies Co., Ltd. Device a22f (rev 21)
```

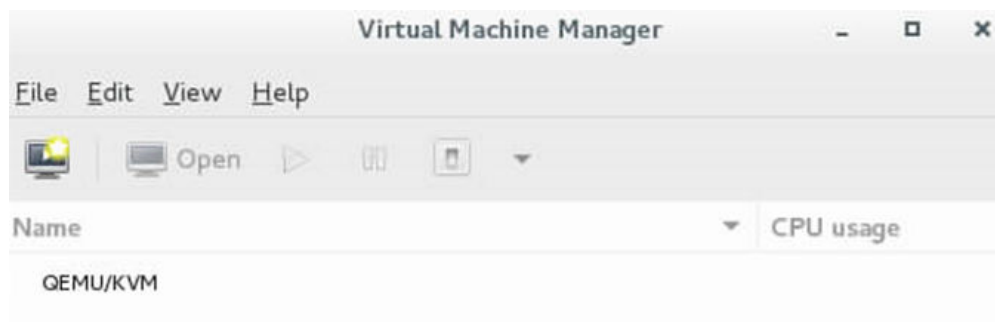
----结束


4.6.2 创建虚拟机

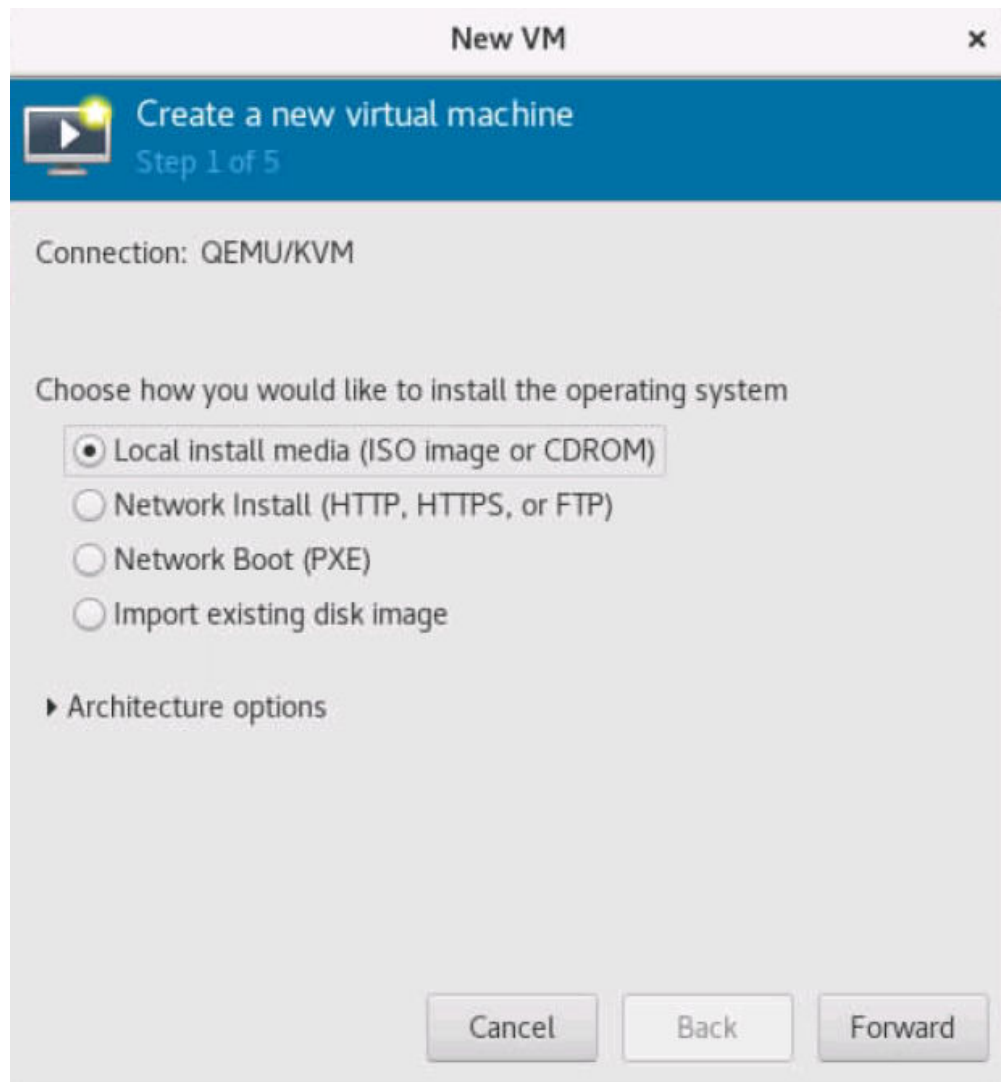
下面以在本地安装Ubuntu 18.04.3虚拟机为例，介绍创建虚拟机的操作步骤。图示中的数据均是示例，具体配置以实际需求为准。本文介绍通过GUI方式和命令行方式创建虚拟机。

GUI 方式创建虚拟机

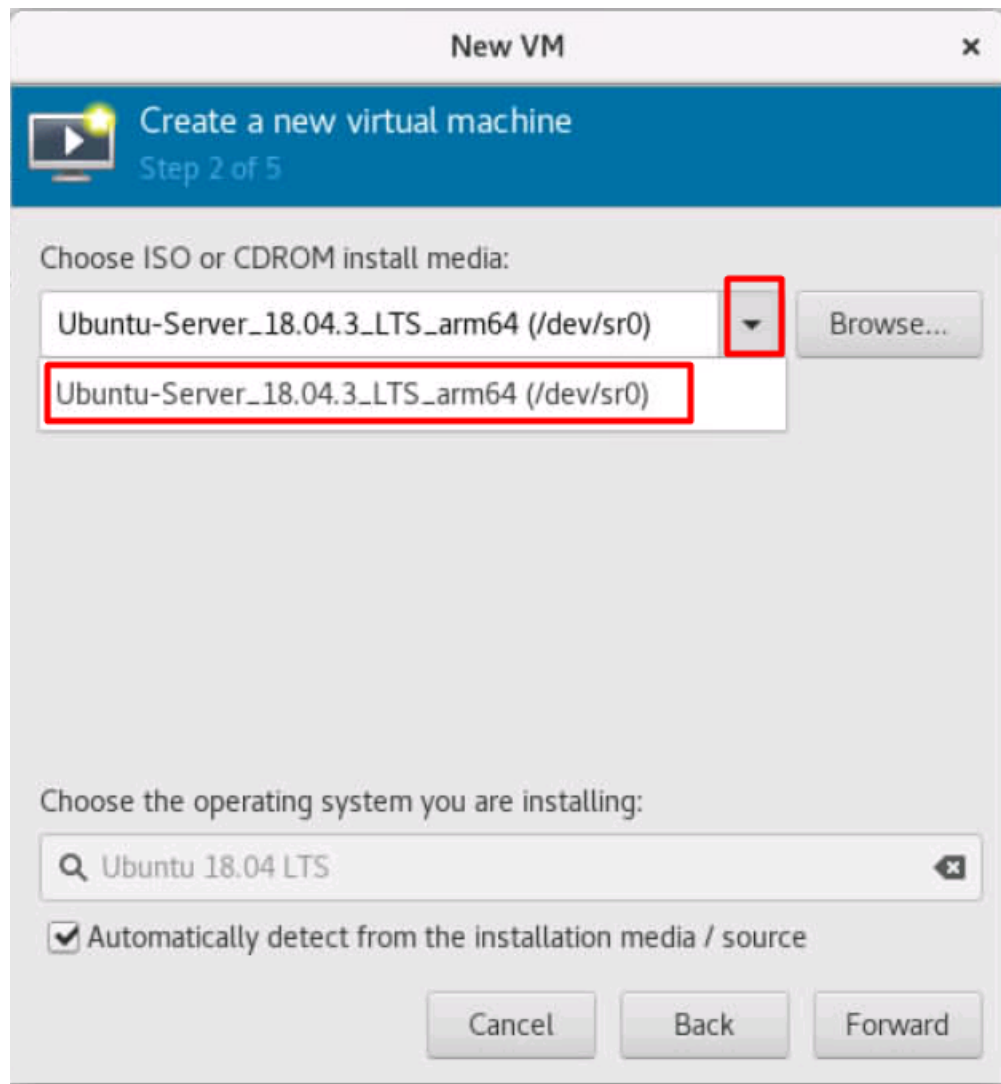
- 步骤1** 打开iBMC远程虚拟控制台，具体操作步骤请参考[A 登录服务器实时桌面](#)。
- 步骤2** 单击“虚拟光驱”，选择“镜像文件”，单击“浏览”，选择客户端本地保存的Ubuntu 18.04.3安装源镜像ISO文件，单击“连接”，将其挂载到虚拟光驱。
- 步骤3** 以root用户登录服务器主机操作系统，在桌面右键鼠标，选择“Open Terminal”打开命令行，在Terminal中执行**virt-manager**命令，进入虚拟机管理界面。



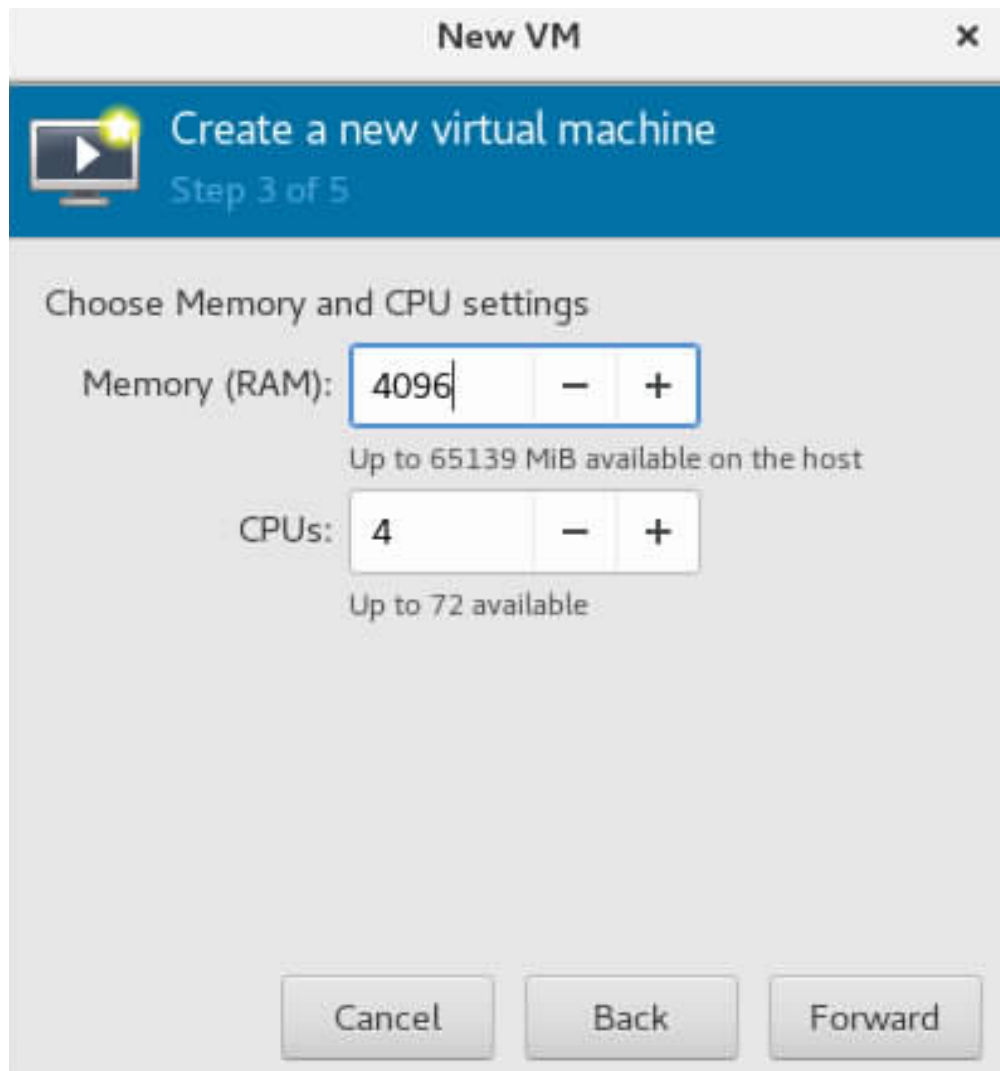
- 步骤4** 单击  创建虚拟机，选择安装系统方式为“Local install media (ISO image or CDROM)”，单击“Forward”。



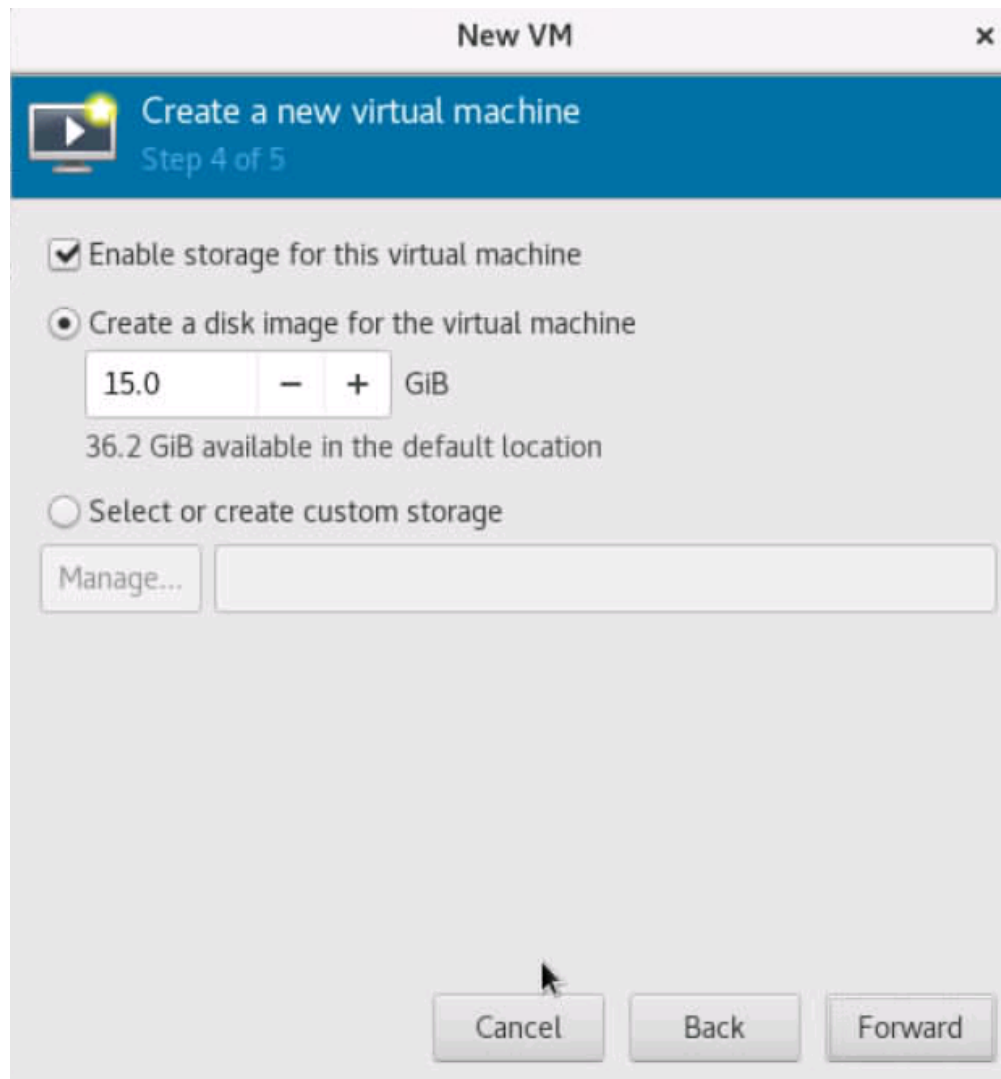
步骤5 单击输入框右侧的 ▾，选择挂载的虚拟光驱（如/dev/sr0），单击“Forward”。



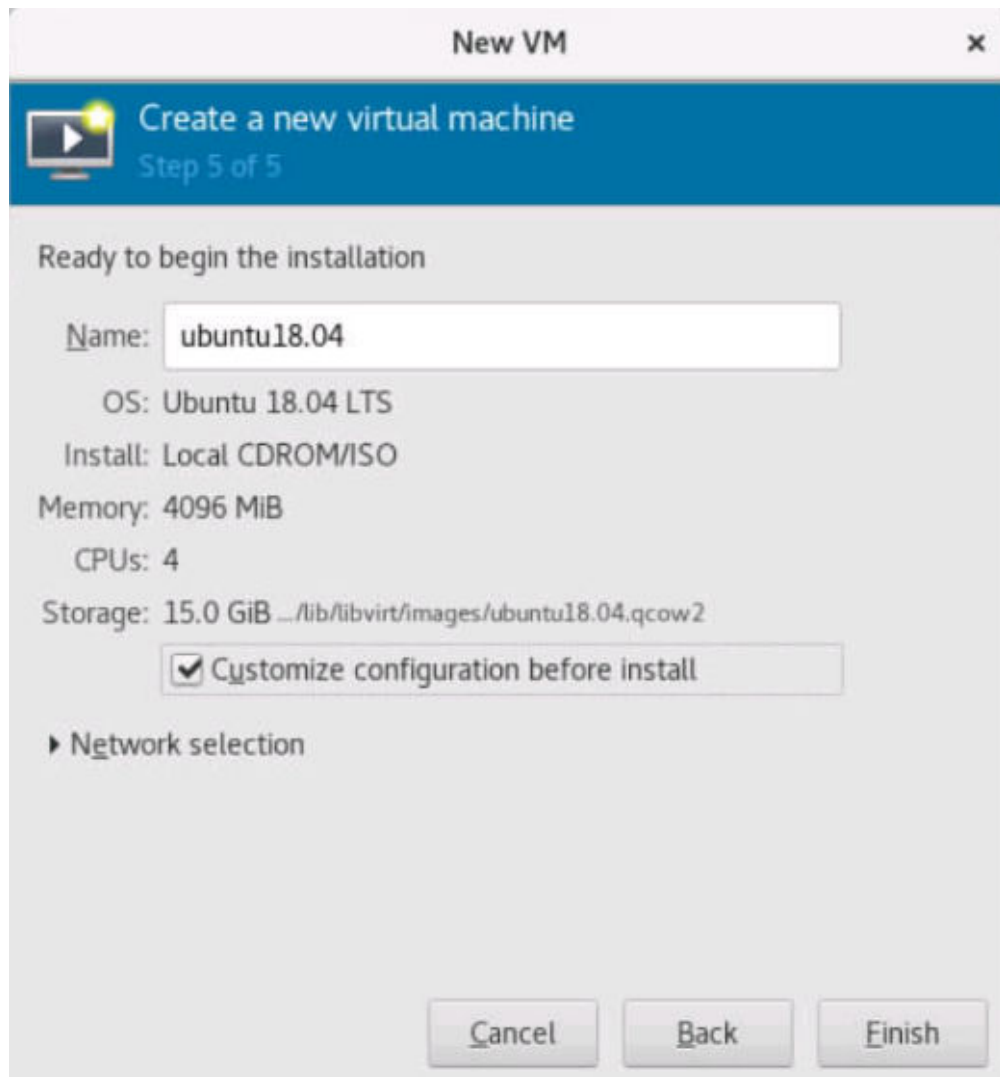
步骤6 配置虚拟机内存大小（单位MB）和CPU数量，单击“Forward”。



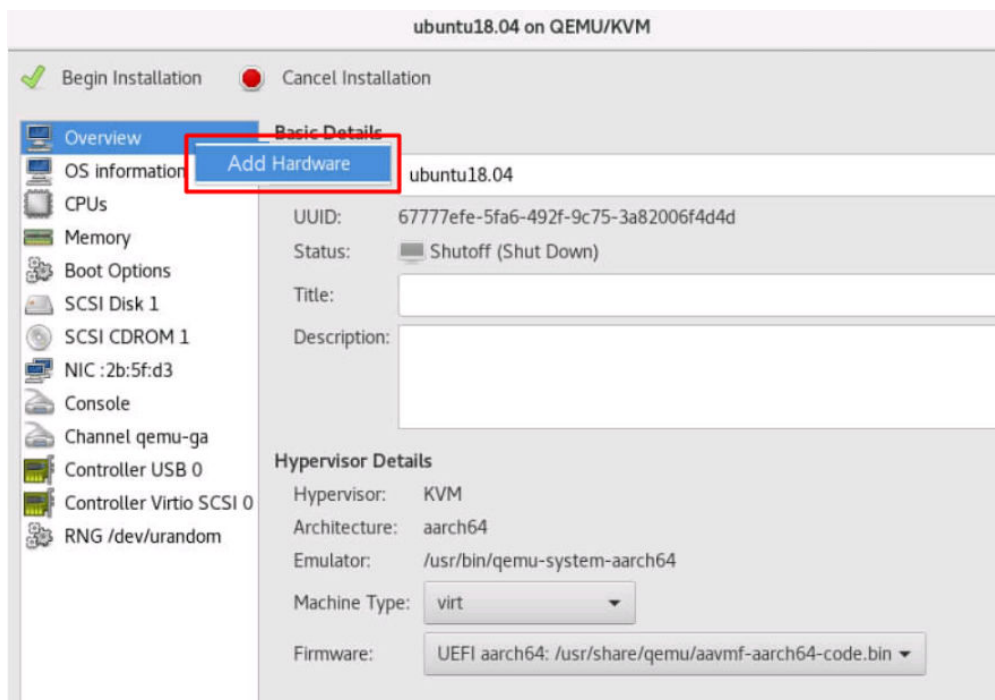
步骤7 配置虚拟机硬盘容量及虚拟机文件存储路径，单击“Forward”。



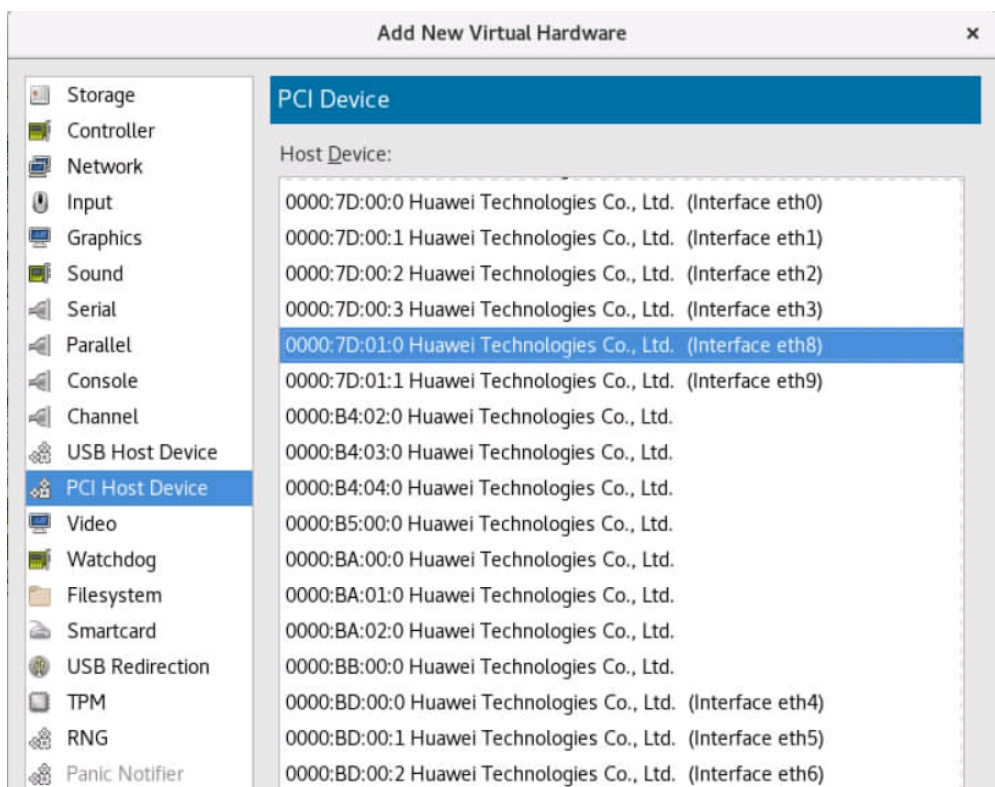
步骤8 勾选“Customize configuration before install”，单击“Finish”。



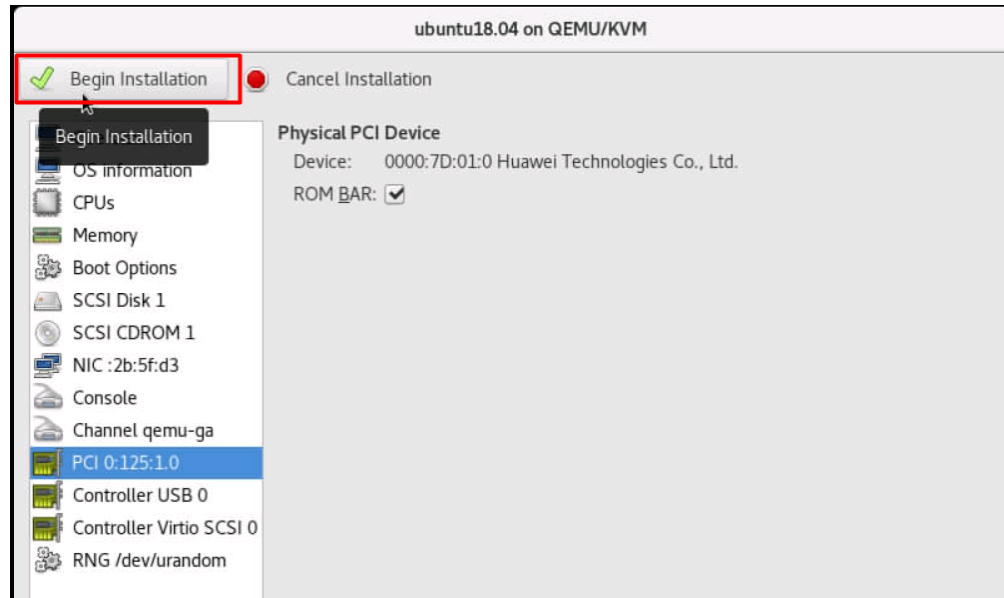
步骤9 在左侧配置项区域任意位置单击鼠标右键，选择“Add Hardware”。



步骤10 在左侧视图选择“PCI Host Device”，在右侧视图选择生成的虚拟化网口设备（即 VF），单击“Finish”添加VF。



步骤11 单击“Begin Installation”安装虚拟机操作系统，操作系统详细安装过程可参考《[华为服务器 操作系统 安装指导 案例集](#)》进行操作。



----结束

命令行方式创建虚拟机

- 步骤1** 以root用户登录服务器OS，在桌面单击鼠标右键，选择“Open Terminal”打开命令行。
- 步骤2** 将操作系统镜像文件拷贝到任意目录下，例如：拷贝到/home目录下。
- 步骤3** 运行如下命令进行安装：

```
virt-install --name ubuntu --ram 2048 --vcpus=4 --location=/home/
ubuntu-18.04.3-server-arm64.iso --disk path=/home/disk,size=20
```

其中各个参数含义如下：

- name：创建的虚拟机名称
- ram：分配给虚拟机使用的内存大小，以MiB为单位
- vcpus：分配给虚拟机使用的CPU个数
- location：安装虚拟机使用的操作系统镜像的路径
- disk：安装虚拟机系统使用的硬盘，path表示硬盘的存放路径，size表示硬盘的大小

----结束

4.6.3 添加 SR-IOV 网口

如果在虚拟机创建过程中未添加虚拟化网卡设备，也可以在虚拟机创建完成后手动添加。

下面以Ubuntu 18.04.3虚拟机为例，介绍添加SR-IOV网口的方法。图示中的数据均是示例，具体配置以实际需求为准。本文介绍通过GUI方式和命令行方式添加SR-IOV网口。

GUI 方式添加 SR-IOV 网口

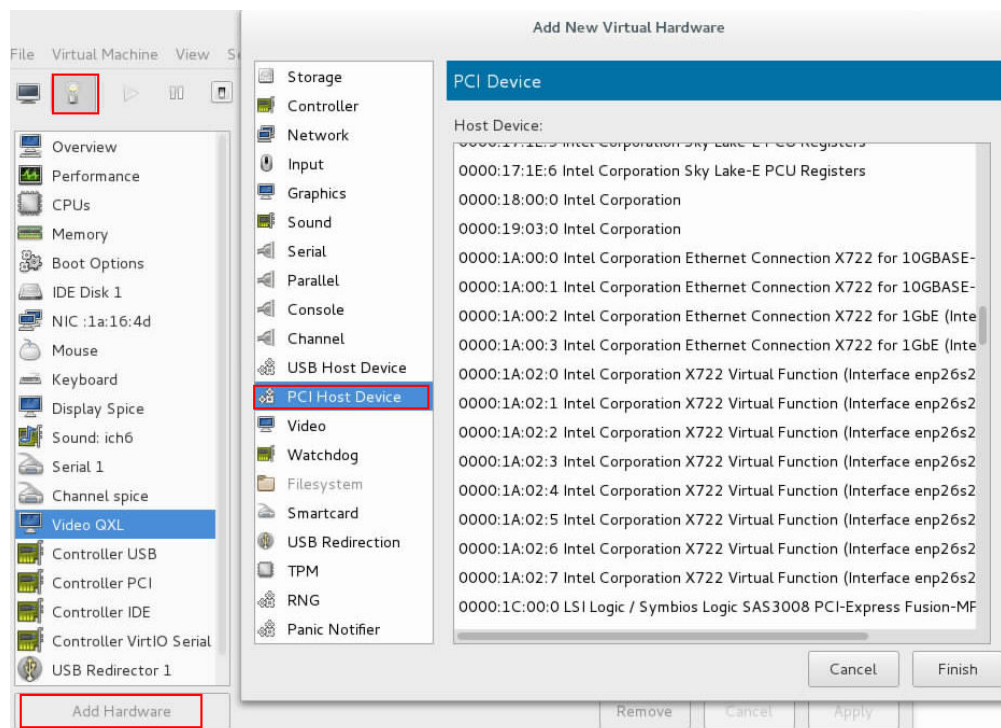
步骤1 完成虚拟机的创建后，以root用户登录虚拟机OS，在桌面单击鼠标右键，选择“Open Terminal”打开命令行。

步骤2 执行如下命令下电虚拟机。

poweroff

步骤3 在新建的虚拟机配置项下选择“Add Hardware > PCI Host Device”，把创建的VF添加到VM中，启动虚拟机，新建的VM中就有VF对应的PCIe设备，如图4-1所示。

图 4-1 把创建的 VF 添加到虚拟机中



步骤4 在虚拟机操作系统中使用如下命令查看新增加的VF设备。

lspci | grep -i eth

回显如下：

```
huawei@172-24-124-79:~$ lspci | grep -i eth
01:00.0 Ethernet controller: Red Hat, Inc. Virtio network device (rev 01)
05:00.0 Ethernet controller: Huawei Technologies Co., Ltd. HNS RDMA Network Controller (Virtual Function) (rev 21)
huawei@172-24-124-79:~$
```

----结束

命令行方式添加 SR-IOV 网口

步骤1 以root用户登录服务器OS，在桌面单击鼠标右键，选择“Open Terminal”打开命令行。

步骤2 使用以下命令对虚拟机进行下电操作。

virsh shutdown ubuntu

其中ubuntu是在创建虚拟机时指定的虚拟机名称，下文中出现的ubuntu若无特殊说明，均表示创建虚拟机时指定的名称。

步骤3 使用以下命令打开虚拟机配置文件。

virsh edit ubuntu

步骤4 输入i对配置文件进行编辑，在配置文件的<device>和</device>之间添加如下内容：

```
<hostdev mode='subsystem' type='pci' managed='yes'>
  <source>
    <address domain='0x0000' bus='0x81' slot='0x01' function='0x2' />
  </source>
</hostdev>
```

其中，domain、bus、slot和function为网卡在系统下的bdf号，可通过lspci查询

步骤5 按“Esc”，输入“:wq!”保存配置文件，并使用以下命令对虚拟机进行上电操作：

virsh start ubuntu

----结束

A 登录服务器实时桌面

您可以通过服务器的高密接口直接连接KVM来作为登录终端，也可以通过服务器iBMC提供的远程控制台进行远程登录。

下面仅介绍通过远程控制台登录服务器实时桌面的操作方法。

步骤1 配置登录环境。

1. 使用网线将PC机网口与服务器的iBMC管理网口相连。
2. 设置PC的IP地址与服务器iBMC管理网口的IP地址在同一网段。
例如：设置IP地址为“192.168.2.10”，子网掩码为“255.255.255.0”。

步骤2 登录服务器iBMC Web界面。

1. 打开浏览器，在地址栏中输入“<http://iBMC管理网口的IP地址>”，按“Enter”。
iBMC登录界面

欢迎到访

iBMC

TaiShan 200 (Model 1280)
huawei



用户名

请输入用户名

密码

请输入密码

域名

这台iBMC

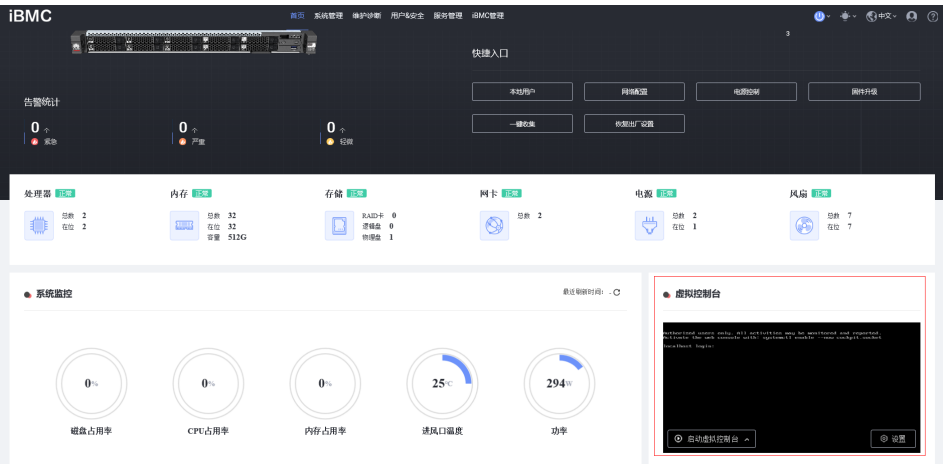
登录


2. 在登录界面中，进行如下设置。
 - a. 输入用户名。
 - b. 输入密码。
 - c. 选择“域名”为“这台iBMC”。
 - d. 单击“登录”。进入iBMC Web主界面。

步骤3 进入虚拟远程控制台。

1. 在“首页”首页右下角选择“虚拟控制台”，如[图A-1](#)所示。

图 A-1 虚拟控制台界面



2. 单击“启动虚拟控制台”右侧的，选择“HTML5集成远程控制台(独占)”、“HTML5集成远程控制台(共享)”、“Java集成远程虚拟控制台(独占)”或“Java集成远程虚拟控制台(共享)”进入服务器的实时操作控制台。
- 已集成HTML5，单击“HTML5集成远程控制台(独占)”或“HTML5集成远程控制台(共享)”。
 - 已集成Java，单击“Java集成远程虚拟控制台(独占)”或“Java集成远程虚拟控制台(共享)”。

----结束

B

缩略语

B		
BMC	Baseboard Management Controller	主板管理控制单元
D		
DPDK	Data Plane Development Kit	数据平面开发套件
E		
ETS	Enhanced Transmission Selection	增强型传送选择
GRO	Generic Receive Offload	通用接收卸载
I		
IP	Internet Protocol	因特网协议
M		
MAC	Media Access Control	媒体接入控制
MTU	Maximum Transmission Unit	最大传输单元
N		
NIC	Network Interface Card	网络接口卡
O		
OS	Operating System	操作系统
P		
PCIe	PCI Express	快速外围元件互连

PF	Physical Function	物理功能
PXE	Preboot Execution Environment	预启动执行环境
R		
RDMA	Remote Direct Memory Access	远端直接内存访问
RSS	Receive Side Scaling	接收端调节
S		
SR-IOV	Single Root I/O Virtualization	单主（根）输入输出虚拟化
T		
TCP	Transmission Control Protocol	传输控制协议
TSO	TCP Segmentation Offload	传输控制协议分片卸载
U		
UDP	User Datagram Protocol	用户数据报协议
V		
VF	Virtual Function	虚拟功能
VLAN	Virtual Local Access Network	虚拟局域网
VM	Virtual Machine	虚拟机
VxLAN	Virtual extensible Local Area Network	虚拟扩展局域网