一、物理机安装依赖包

apt install gconf2 qemu-system qemu-system-arm qemu-utils qemu-efi libvirt-daemon-system libvirt-clients bridge-utils virtinst virt-manager seabios vgabios gir1.2-spiceclientgtk-3.0 xauth

x11 字库（可选）
apt install fonts-noto*

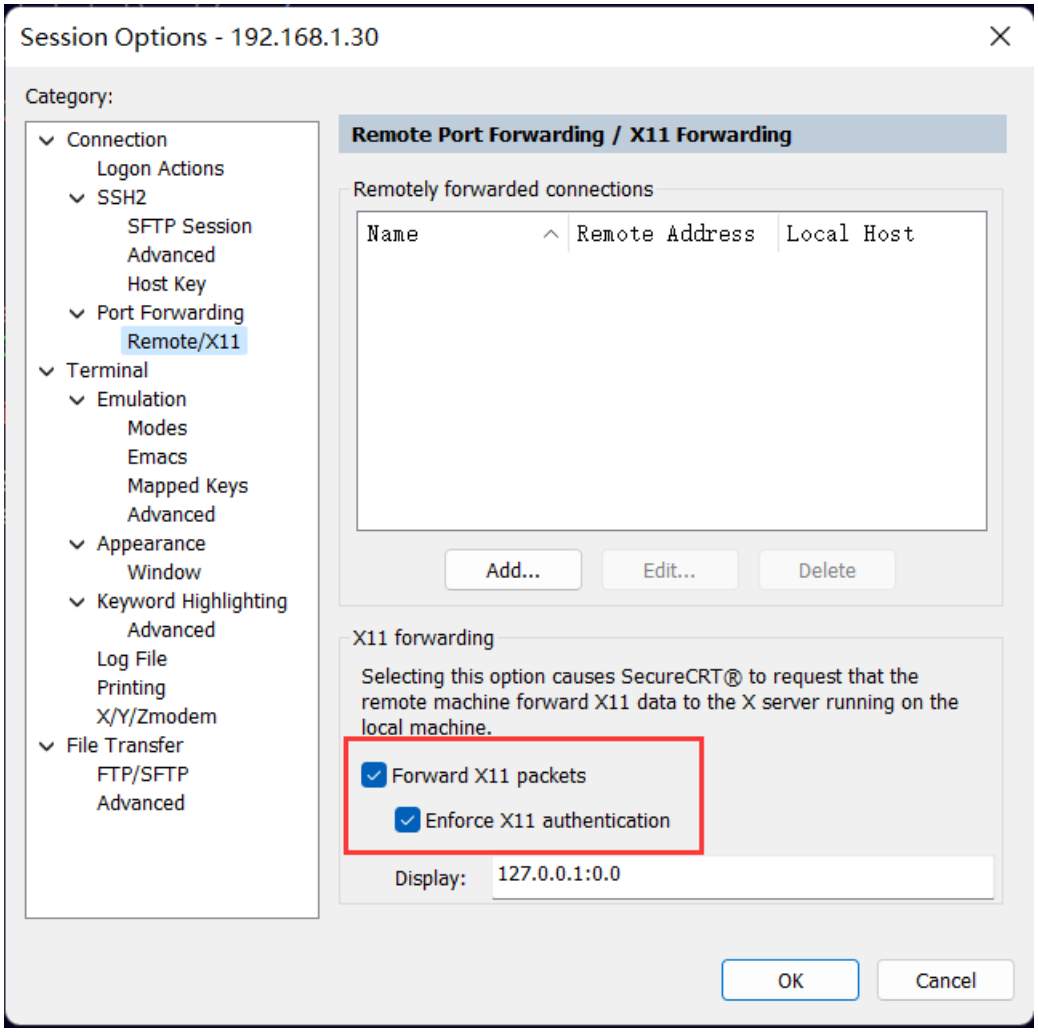
桌面环境(可选)
apt install tasksel
运行 tasksel，选择至少一个桌面环境即可

二、windows 客户机安装 ssh 客户端及 x11 server

客户端可以用 xshell、securecrt 等等，x11 server 可以用 xshell 自带的，或者 xming、vcxsrv、cygwin x11 等。
以 securecrt+xming 为例：

启动之后，图标在右下角（需要防火墙入站规则允许 xming）

Securecrt session 配置：（开启 forward x11）



三、物理机网络配置

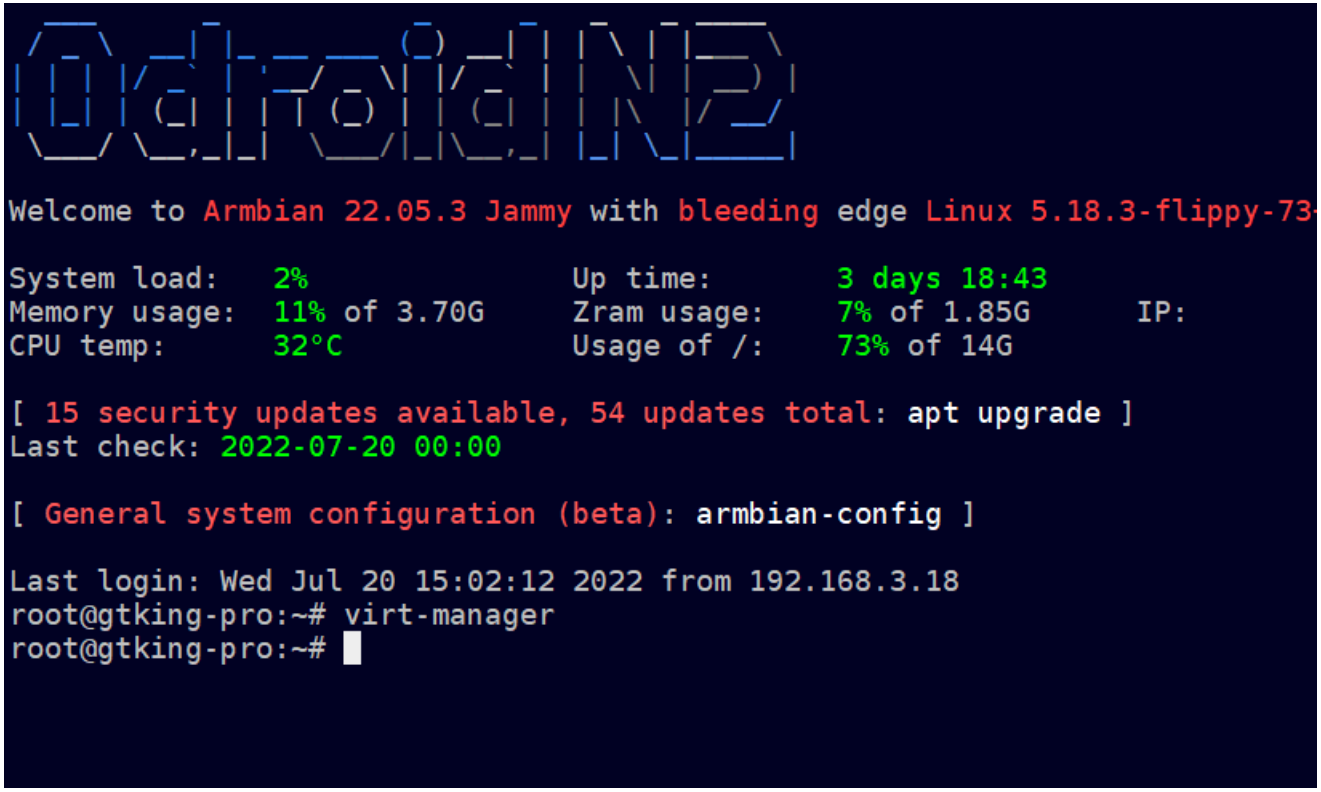
注意：如果物理机只有单网卡的话，要把网络改成桥接，以便与虚机共用网卡。
配置示例：
文件名：/etc/network/interfaces.d/br0

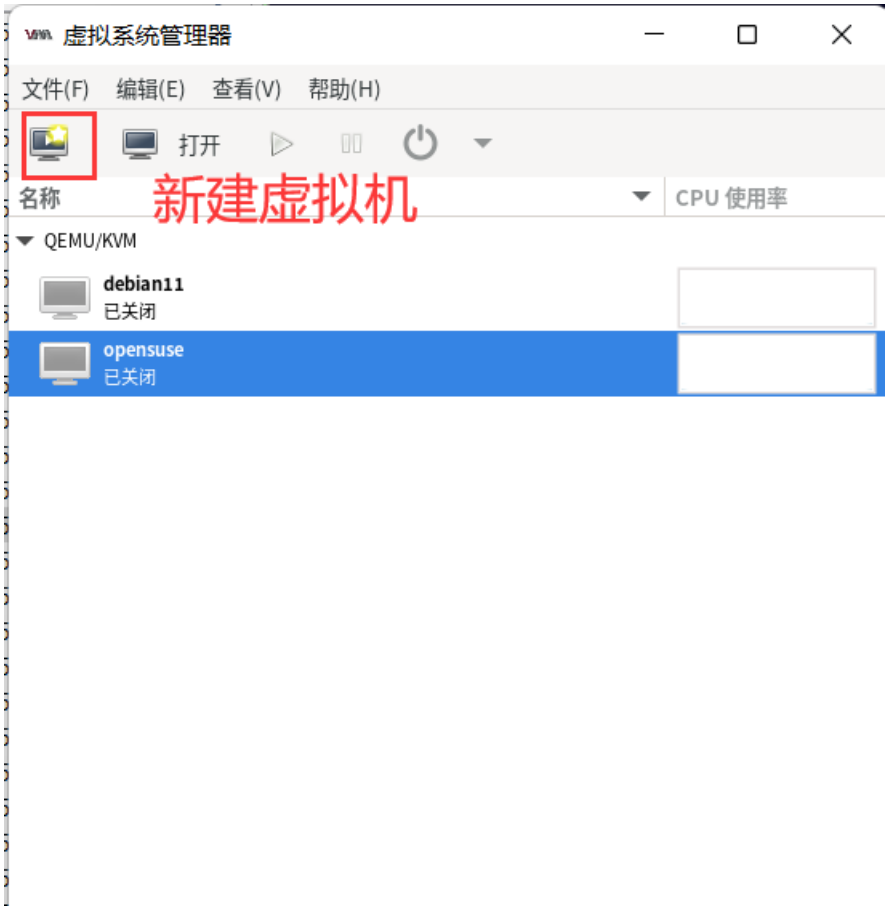
```
# eth0 setup
allow-hotplug eth0
iface eth0 inet manual
  pre-up   ifconfig $IFACE up
  pre-down ifconfig $IFACE down

# Bridge setup
auto br0
iface br0 inet static
  bridge_ports eth0
  bridge_stp off
  bridge_waitport 0
  bridge_fd 0
  address 192.168.3.22
  broadcast 192.168.3.255
  netmask 255.255.255.0
  gateway 192.168.3.1
  dns-nameservers 192.168.3.1
```

四、安装过程截图

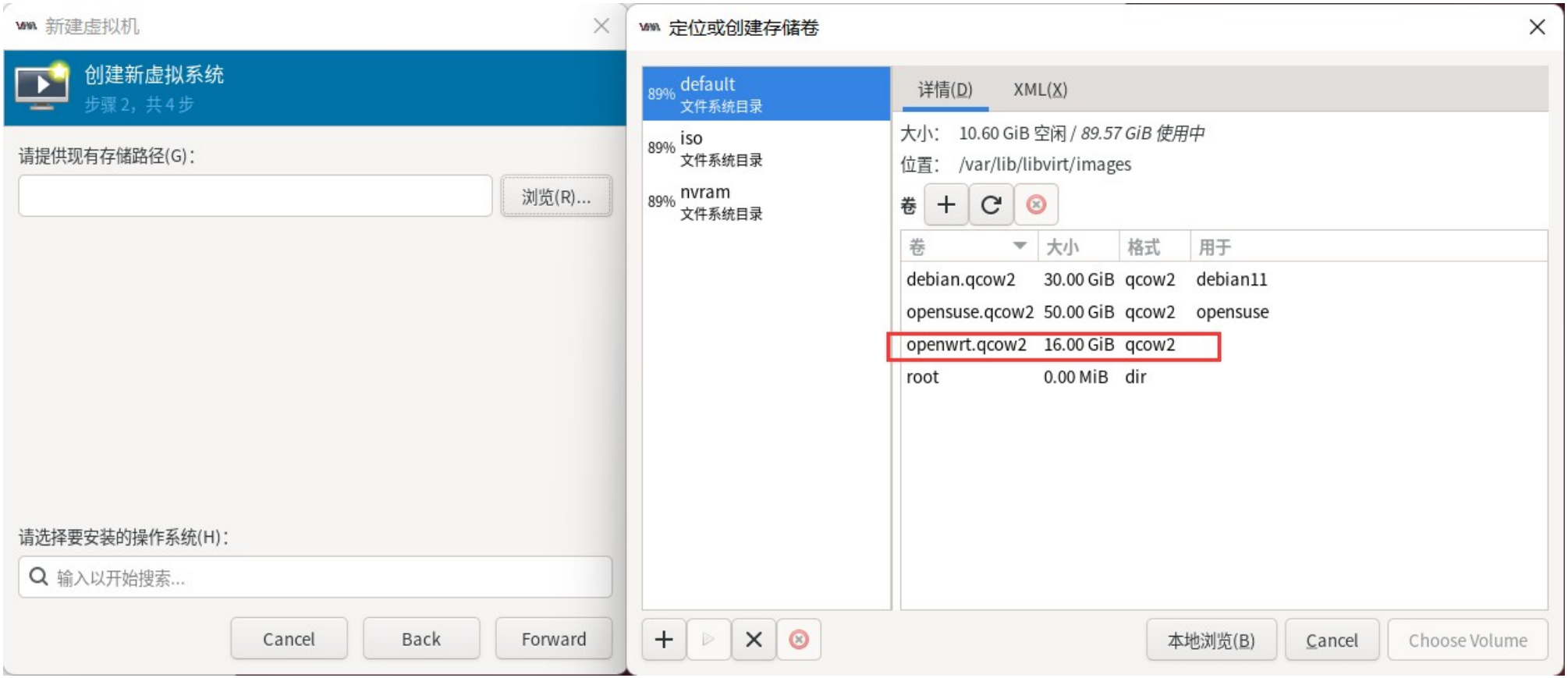
qemu 的固件镜像后缀是 .qcow2，把镜像上传到物理机的 /var/lib/libvirt/images/目录下，名字可以任意改。

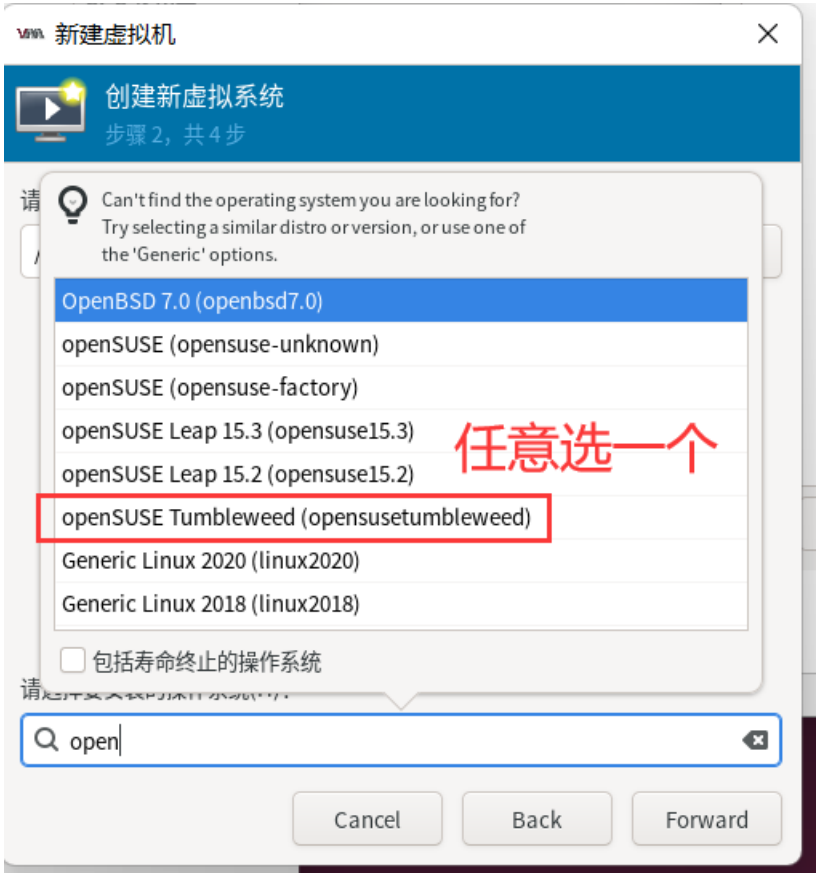
运行 virt-manager （或桌面环境下点击“虚拟机管理”图标）




虚拟系统管理器
文件(F)
编辑(E)
查看(V)
帮助(H)
打开
新建虚拟机
名称
CPU 使用率
QEMU/KVM
debian11
已关闭
opensuse
已关闭

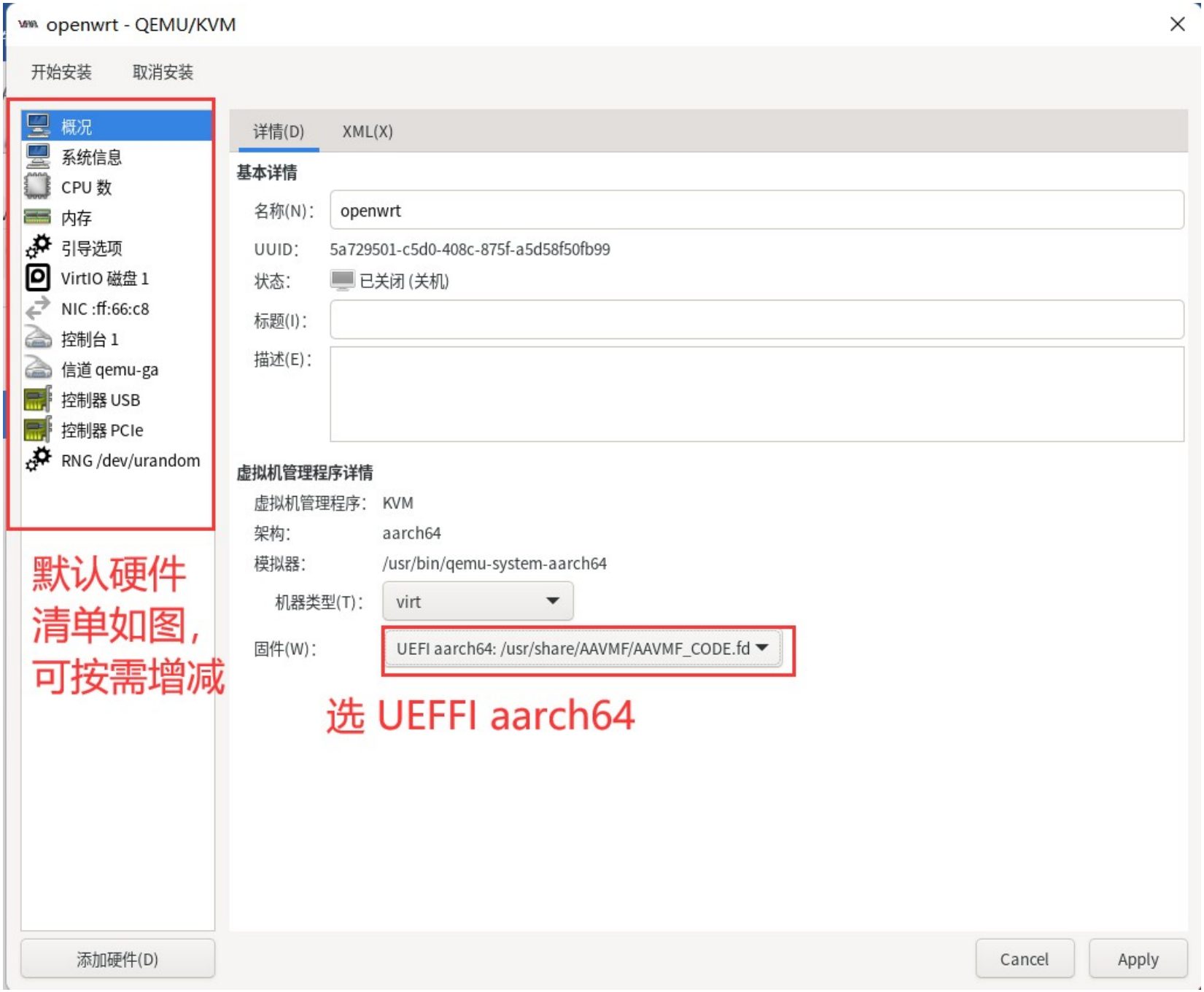

新建虚拟机
创建新虚拟系统
步骤 1，共 4 步
连接(O)： QEMU/KVM
选择如何安装操作系统
本地安装介质(ISO 映像或者光驱)(L)
网络安装（HTTP、HTTPS 或 FTP）(I)
导入现有磁盘映像(E)
手动安装(N)
架构选项
虚拟类型(V)： KVM
架构(A)： aarch64
机器类型(M)： virt
Cancel
Back
Forward

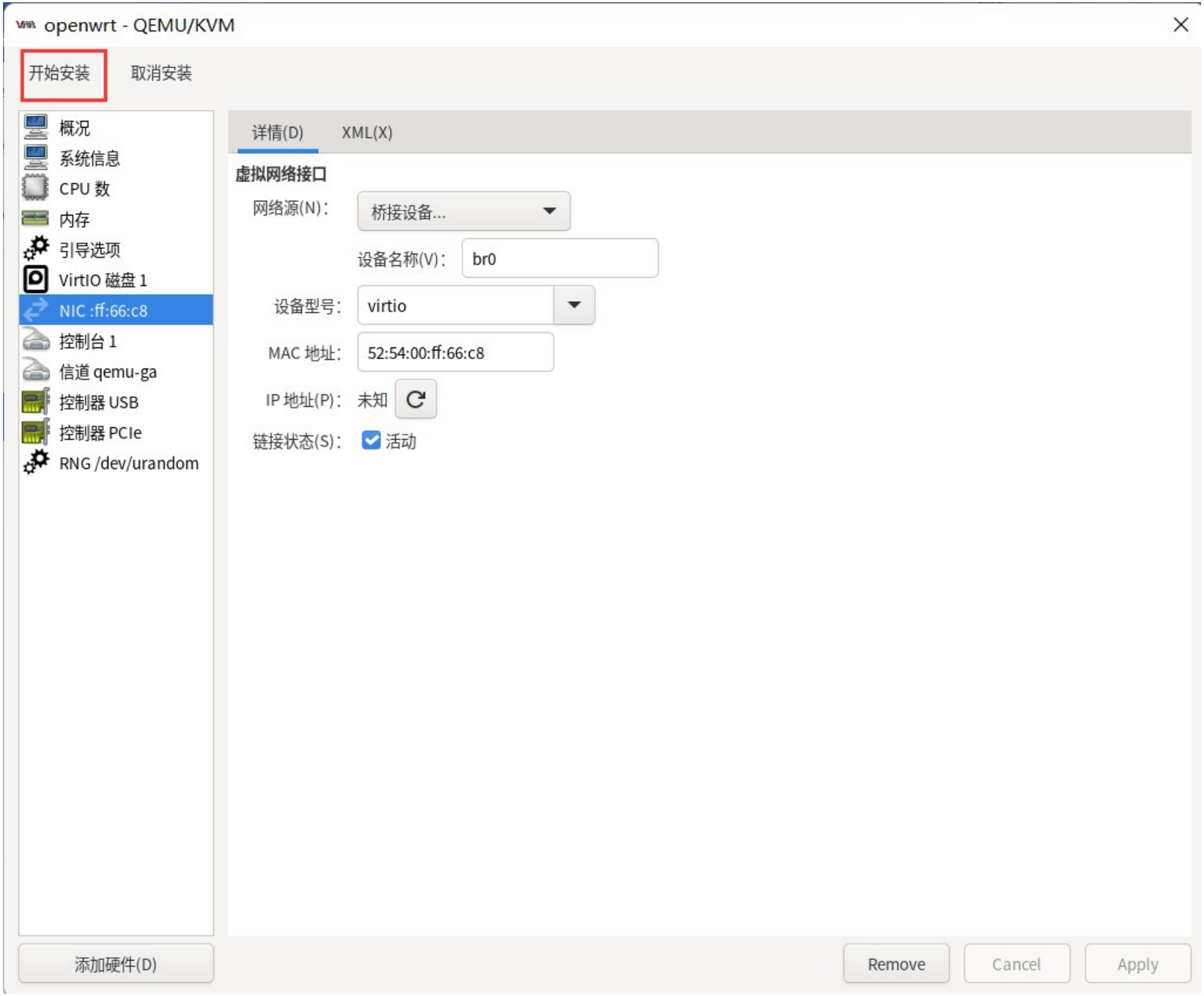

新建虚拟机
创建新虚拟系统
步骤 2，共 4 步
请提供现有存储路径(G)：
浏览(R)...
请选择要安装的操作系统(H)：
输入以开始搜索...
Cancel
Back
Forward
定位或创建存储卷
89% default
文件系统目录
89% iso
文件系统目录
89% nvram
文件系统目录
详情(D)
XML(X)
大小： 10.60 GiB 空闲 / 89.57 GiB 使用中
位置： /var/lib/libvirt/images
卷
卷
大小
格式
用于
debian.qcow2 30.00 GiB qcow2 debian11
opensuse.qcow2 50.00 GiB qcow2 opensuse
openwrt.qcow2 16.00 GiB qcow2
root 0.00 MiB dir
本地浏览(B)
Cancel
Choose Volume

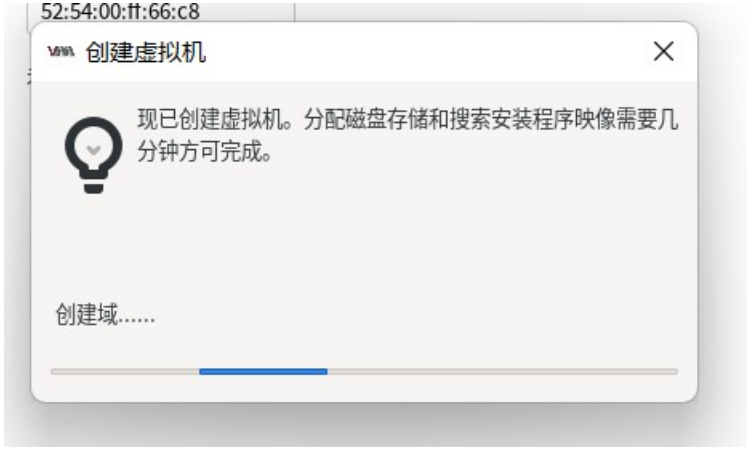

新建虚拟机
创建新虚拟系统
步骤 2，共 4 步
Can't find the operating system you are looking for?
Try selecting a similar distro or version, or use one of
the 'Generic' options.
OpenBSD 7.0 (openbsd7.0)
openSUSE (opensuse-unknown)
openSUSE (opensuse-factory)
openSUSE Leap 15.3 (opensuse15.3)
openSUSE Leap 15.2 (opensuse15.2)
openSUSE Tumbleweed (opensusetumbleweed)
Generic Linux 2020 (linux2020)
Generic Linux 2018 (linux2018)
任意选一个
包括寿命终止的操作系统
open
Cancel
Back
Forward

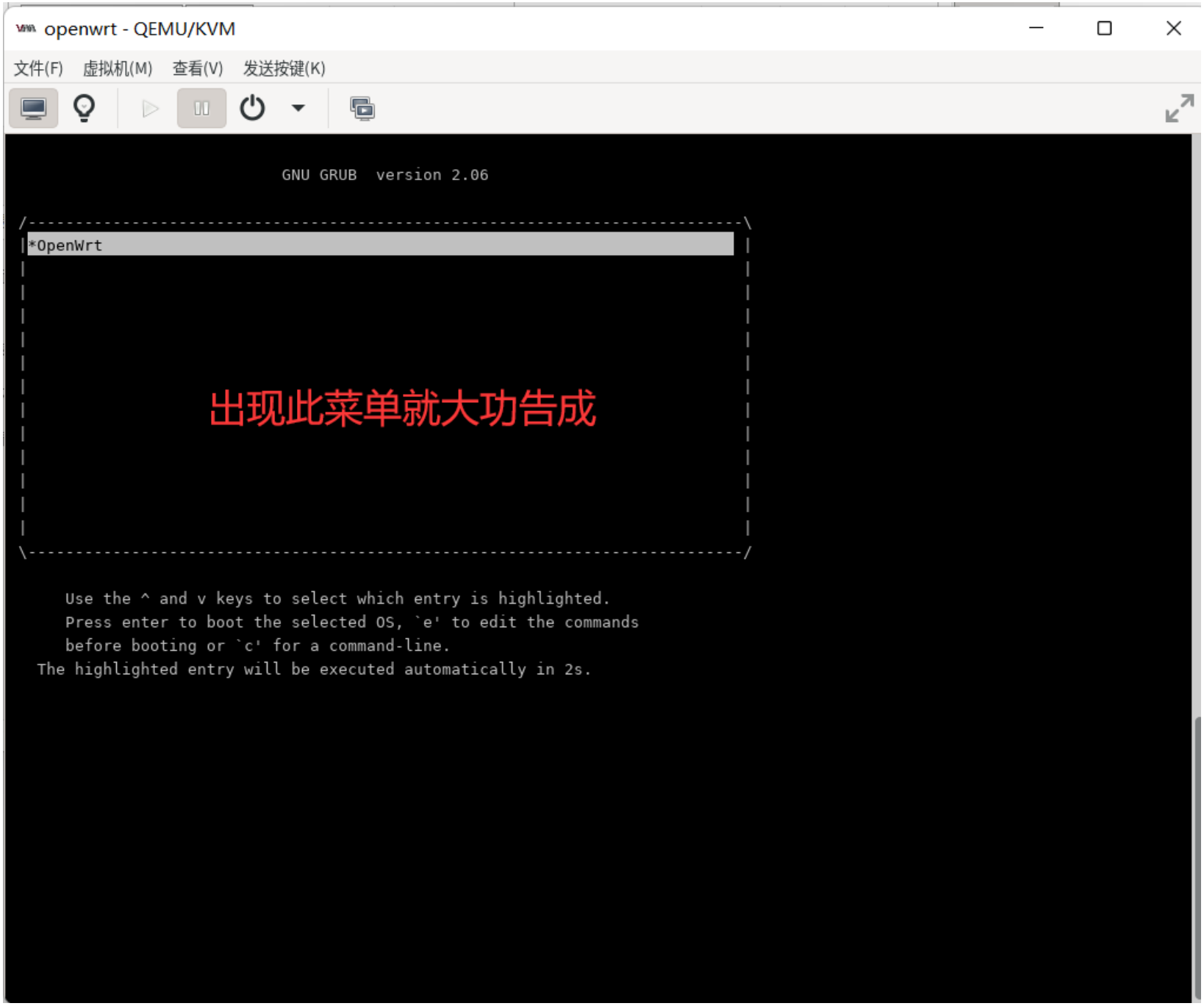

新建虚拟机
创建新虚拟系统
步骤 3，共 4 步
请选择内存和 CPU 设置：
内存(M)：
2048
主机中最多有 3784 MiB 可用
CPU 数(P)：
2
最多可用 6
Cancel
Back
Forward

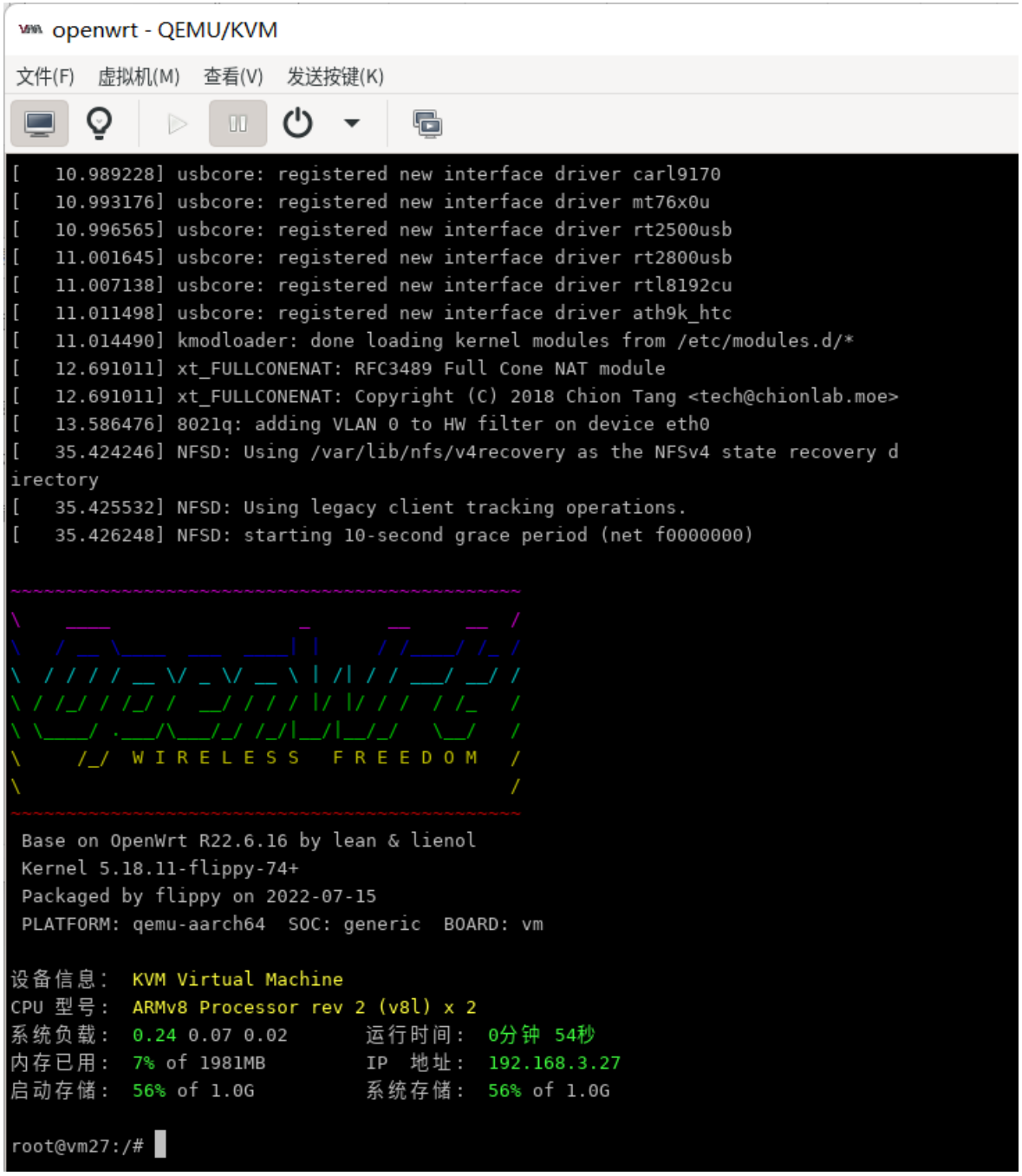

新建虚拟机
创建新虚拟系统
步骤 4，共 4 步
准备开始安装
名称(N)：
openwrt
操作系统：
openSUSE Tumbleweed
安装：
导入现有操作系统映像
内存：
2048 MiB
CPU 数：
2
存储：
…/lib/libvirt/images/openwrt.qcow2
在安装前自定义配置(U)
选择网络(E)
桥接设备...
设备名称(V)：
br0
共享网口选桥接，独立网口选macvtap
Cancel
Back
完成(F)

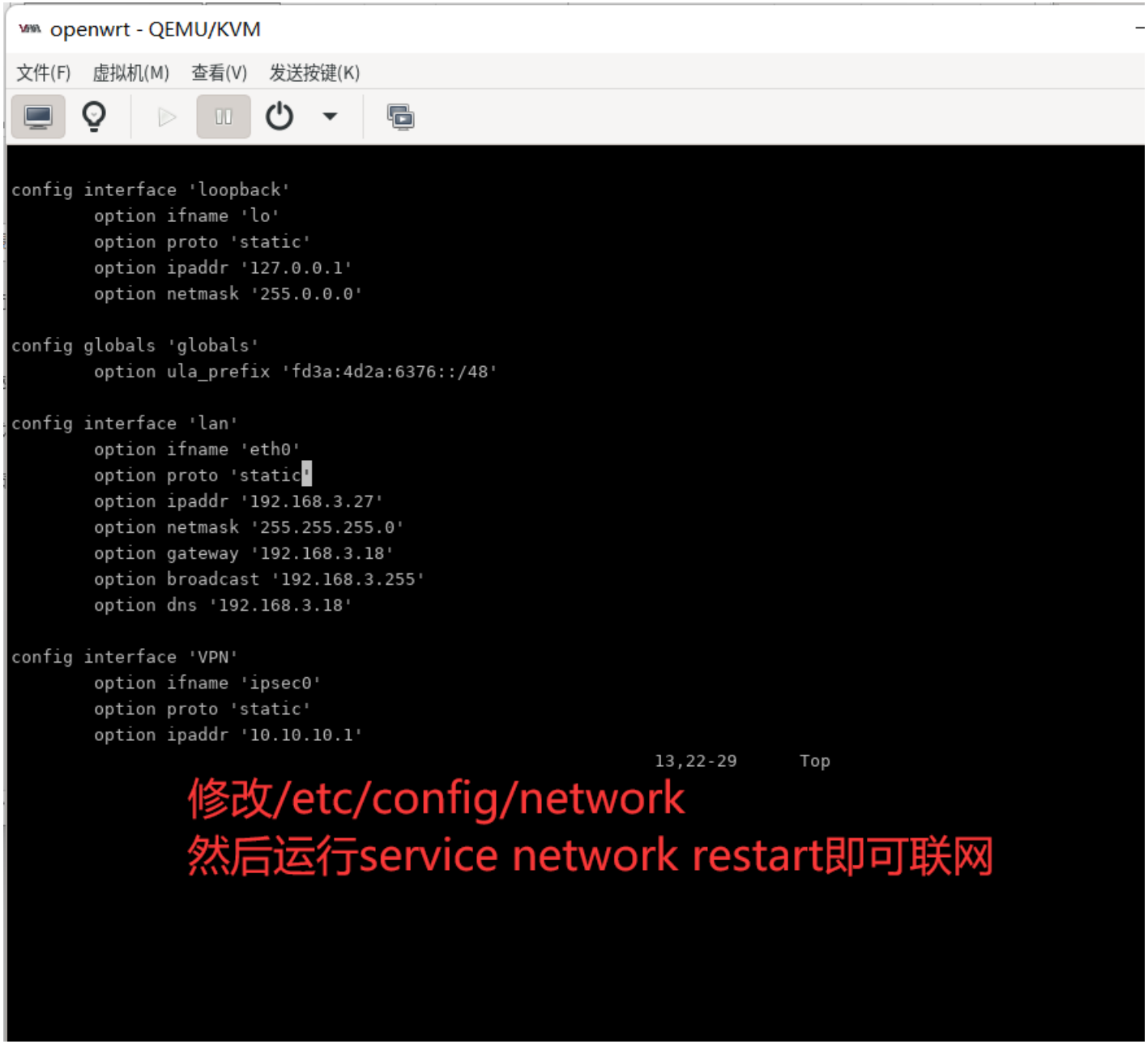

openwrt - QEMU/KVM
开始安装
取消安装
概况
系统信息
CPU 数
内存
引导选项
VirtIO 磁盘 1
NIC :ff:66:c8
控制台 1
信道 qemu-ga
控制器 USB
控制器 PCIe
RNG /dev/urandom
详情(D)
XML(X)
基本详情
名称(N):
openwrt
UUID:
5a729501-c5d0-408c-875f-a5d58f50fb99
状态:
已关闭 (关机)
标题(I):
描述(E):
虚拟机管理程序详情
虚拟机管理程序:
KVM
架构:
aarch64
模拟器:
/usr/bin/qemu-system-aarch64
机器类型(T):
virt
固件(W):
UEFI aarch64: /usr/share/AAVMF/AAVMF_CODE.fd
默认硬件清单如图，可按需增减
选 UEFFI aarch64
添加硬件(D)
Cancel
Apply


openwrt - QEMU/KVM
开始安装
取消安装
概况
系统信息
CPU 数
内存
引导选项
VirtIO 磁盘 1
NIC :ff:66:c8
控制台 1
信道 qemu-ga
控制器 USB
控制器 PCIe
RNG /dev/urandom
详情(D)
XML(X)
虚拟网络接口
网络源(N):
桥接设备...
设备名称(V):
br0
设备型号:
virtio
MAC 地址:
52:54:00:ff:66:c8
IP 地址(P):
未知
链接状态(S):
活动
添加硬件(D)
Remove
Cancel
Apply


52:54:00:ff:66:c8
创建虚拟机
现已创建虚拟机。分配磁盘存储和搜索安装程序映像需要几分钟方可完成。
创建域.....


openwrt - QEMU/KVM
文件(F) 虚拟机(M) 查看(V) 发送按键(K)
GNU GRUB version 2.06
*OpenWrt
出现此菜单就大功告成
Use the ^ and v keys to select which entry is highlighted.
Press enter to boot the selected OS, `e' to edit the commands
before booting or `c' for a command-line.
The highlighted entry will be executed automatically in 2s.


openwrt - QEMU/KVM
文件(F) 虚拟机(M) 查看(V) 发送按键(K)
[10.989228] usbcore: registered new interface driver carl9170
[10.993176] usbcore: registered new interface driver mt76x0u
[10.996565] usbcore: registered new interface driver rt2500usb
[11.001645] usbcore: registered new interface driver rt2800usb
[11.007138] usbcore: registered new interface driver rtl8192cu
[11.011498] usbcore: registered new interface driver ath9k_htc
[11.014490] kmodloader: done loading kernel modules from /etc/modules.d/*
[12.691011] xt_FULLCONENAT: RFC3489 Full Cone NAT module
[12.691011] xt_FULLCONENAT: Copyright (C) 2018 Chion Tang <tech@chionlab.moe>
[13.586476] 8021q: adding VLAN 0 to HW filter on device eth0
[35.424246] NFSD: Using /var/lib/nfs/v4recovery as the NFSv4 state recovery directory
[35.425532] NFSD: Using legacy client tracking operations.
[35.426248] NFSD: starting 10-second grace period (net f0000000)
WIRELESS FREEDOM
Base on OpenWrt R22.6.16 by lean & lienol
Kernel 5.18.11-flippy-74+
Packaged by flippy on 2022-07-15
PLATFORM: qemu-aarch64 SOC: generic BOARD: vm
设备信息: KVM Virtual Machine
CPU 型号: ARMv8 Processor rev 2 (v8l) x 2
系统负载: 0.24 0.07 0.02
运行时间: 0分钟 54秒
内存已用: 7% of 1981MB
IP 地址: 192.168.3.27
启动存储: 56% of 1.0G
系统存储: 56% of 1.0G
root@vm27:/#


openwrt - QEMU/KVM
文件(F) 虚拟机(M) 查看(V) 发送按键(K)
config interface 'loopback'
option ifname 'lo'
option proto 'static'
option ipaddr '127.0.0.1'
option netmask '255.0.0.0'
config globals 'globals'
option ula_prefix 'fd3a:4d2a:6376::/48'
config interface 'lan'
option ifname 'eth0'
option proto 'static'
option ipaddr '192.168.3.27'
option netmask '255.255.255.0'
option gateway '192.168.3.18'
option broadcast '192.168.3.255'
option dns '192.168.3.18'
config interface 'VPN'
option ifname 'ipsec0'
option proto 'static'
option ipaddr '10.10.10.1'
13,22-29 Top
修改/etc/config/network
然后运行service network restart即可联网

vm27®

- 状态
  - 概览
  - 防火墙
  - 路由表
  - 系统日志
  - 内核日志
  - 系统进程
  - 实时信息
  - 实时监控
  - WireGuard 状态
  - 负载均衡
  - 释放内存
- 系统
- 服务
- Docker
- 网络存储
- VPN

状态

系统

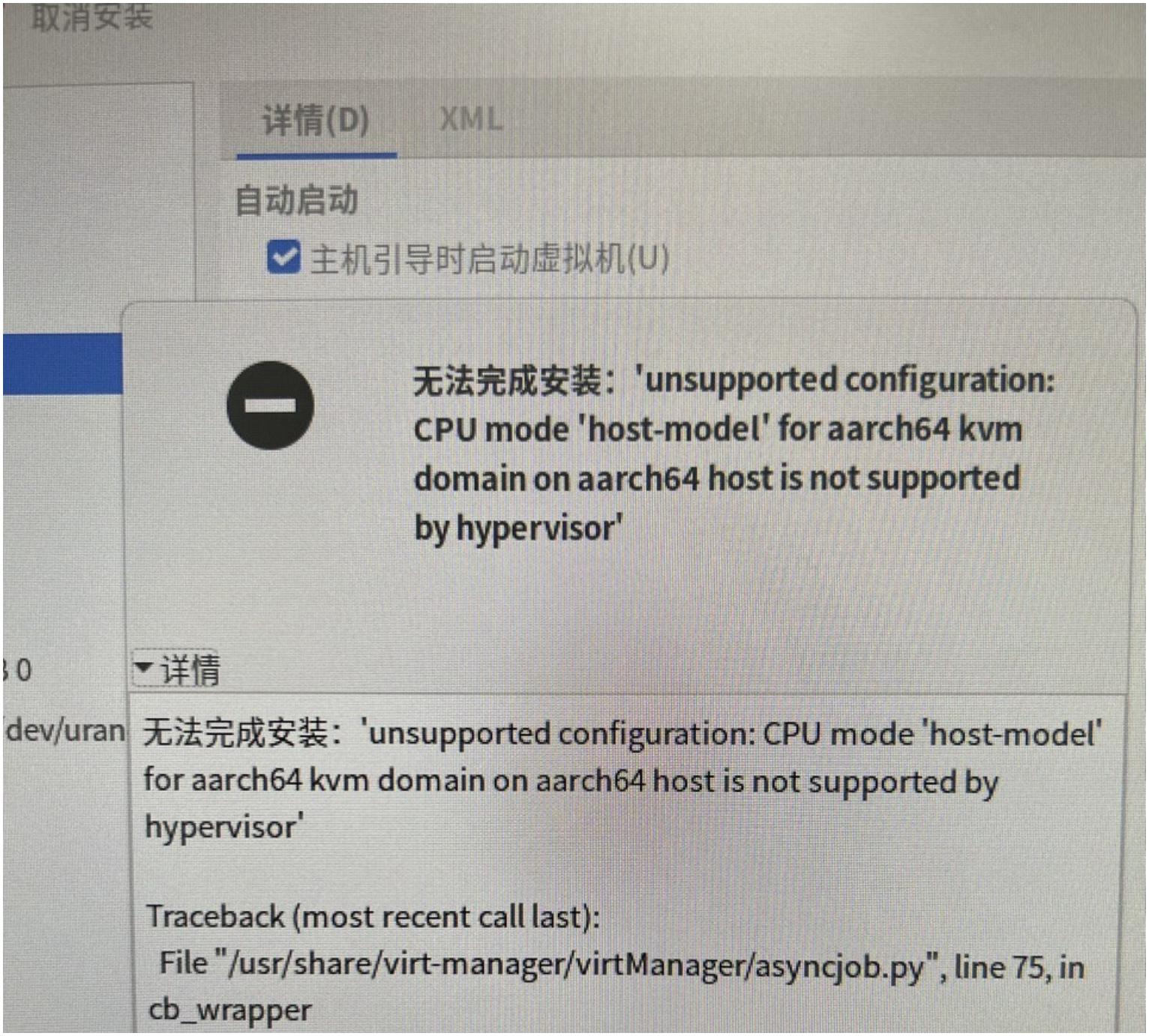
主机名	vm27	
型号	KVM Virtual Machine	
	CPU CoreMark	24876.008644
	aes-128-gcm(1K)	2274377.73k
	aes-256-gcm(1K)	1942573.06k
	chacha20-poly1305(1K)	732791.47k
架构	ARMv8 Processor rev 2 x 2	
固件版本	OpenWrt R22.6.16 (2022-07-15 23:27:16 by flippy) / LuCI Master (git-22.193.59890-c3	
内核版本	5.18.11-flippy-74+	
本地时间	Wed Jul 20 15:39:25 2022	
运行时间	0h 2m 29s	
平均负载	0.23, 0.10, 0.03	
CPU 使用率（%）	5 %	

内存

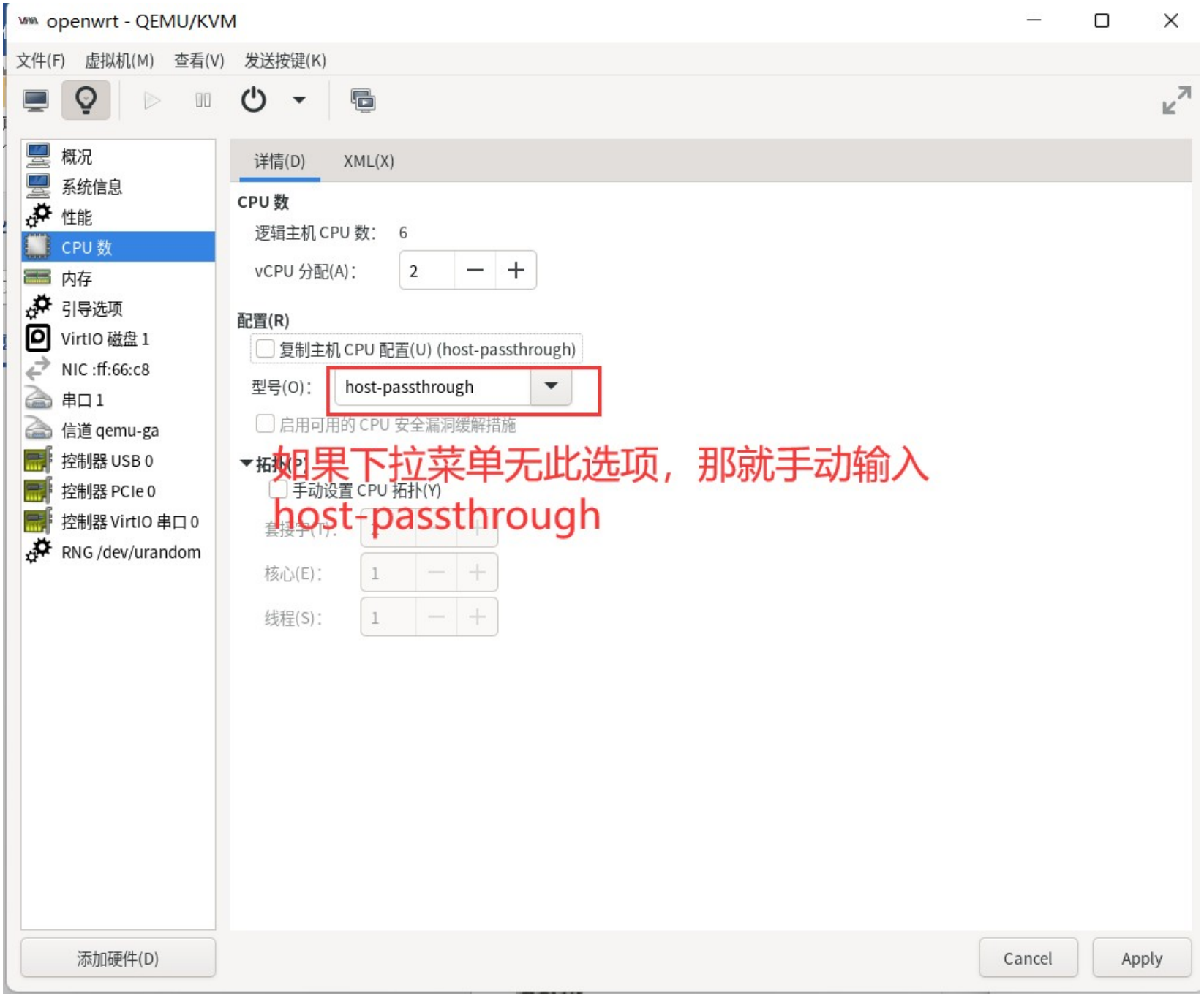
可用数	1801 MB / 19
已缓存	2 MB / 198

五、故障处理

5.1 cpu 模式不对：提示 cpu mode ‘host-mode’ not supported


取消安装
详情(D)
XML
自动启动
主机引导时启动虚拟机(U)
无法完成安装：'unsupported configuration: CPU mode 'host-model' for aarch64 kvm domain on aarch64 host is not supported by hypervisor'
详情
无法完成安装：'unsupported configuration: CPU mode 'host-model' for aarch64 kvm domain on aarch64 host is not supported by hypervisor'
Traceback (most recent call last):
File "/usr/share/virt-manager/virtManager/asyncjob.py", line 75, in cb_wrapper

解决方法：把 cpu 模式手动改成 host-passthrough (如果下拉选项没这个，那就手动输入)


openwrt - QEMU/KVM
文件(F)
虚拟机(M)
查看(V)
发送按键(K)
概况
系统信息
性能
CPU 数
内存
引导选项
VirtIO 磁盘 1
NIC :ff:66:c8
串口 1
信道 qemu-ga
控制器 USB 0
控制器 PCIe 0
控制器 VirtIO 串口 0
RNG /dev/urandom
添加硬件(D)
详情(D)
XML(X)
CPU 数
逻辑主机 CPU 数: 6
vCPU 分配(A): 2
配置(R)
复制主机 CPU 配置(U) (host-passthrough)
型号(O): host-passthrough
启用可用的 CPU 安全漏洞缓解措施
拓扑(P)
手动设置 CPU 拓扑(Y)
套接字(T): 1
核心(E): 1
线程(S): 1
如果下拉菜单无此选项，那就手动输入
host-passthrough
Cancel
Apply

5.2 虚拟机服务未启动

systemctl status libvirtd

正常情况应该这样：

```
root@gtking-pro:/etc/libvirt/qemu# systemctl status libvirtd
● libvirtd.service - Virtualization daemon
     Loaded: loaded (/lib/systemd/system/libvirtd.service; enabled; vendor preset: enabled)
     Active: active (running) since Sat 2022-07-16 20:45:31 CST; 4 days ago
TriggeredBy: ● libvirtd.socket
             ● libvirtd-admin.socket
             ● libvirtd-ro.socket
       Docs: man:libvirtd(8)
             https://libvirt.org
   Main PID: 2087 (libvirtd)
      Tasks: 23 (limit: 32768)
     Memory: 175.9M
        CPU: 13.546s
     CGroup: /system.slice/libvirtd.service
             ├─2087 /usr/sbin/libvirtd
             ├─2497 /usr/sbin/dnsmasq --conf-file=/var/lib/libvirt/dnsmasq/default.conf --leasefile-ro --dhcp-sc
             └─2503 /usr/sbin/dnsmasq --conf-file=/var/lib/libvirt/dnsmasq/default.conf --leasefile-ro --dhcp-sc
```

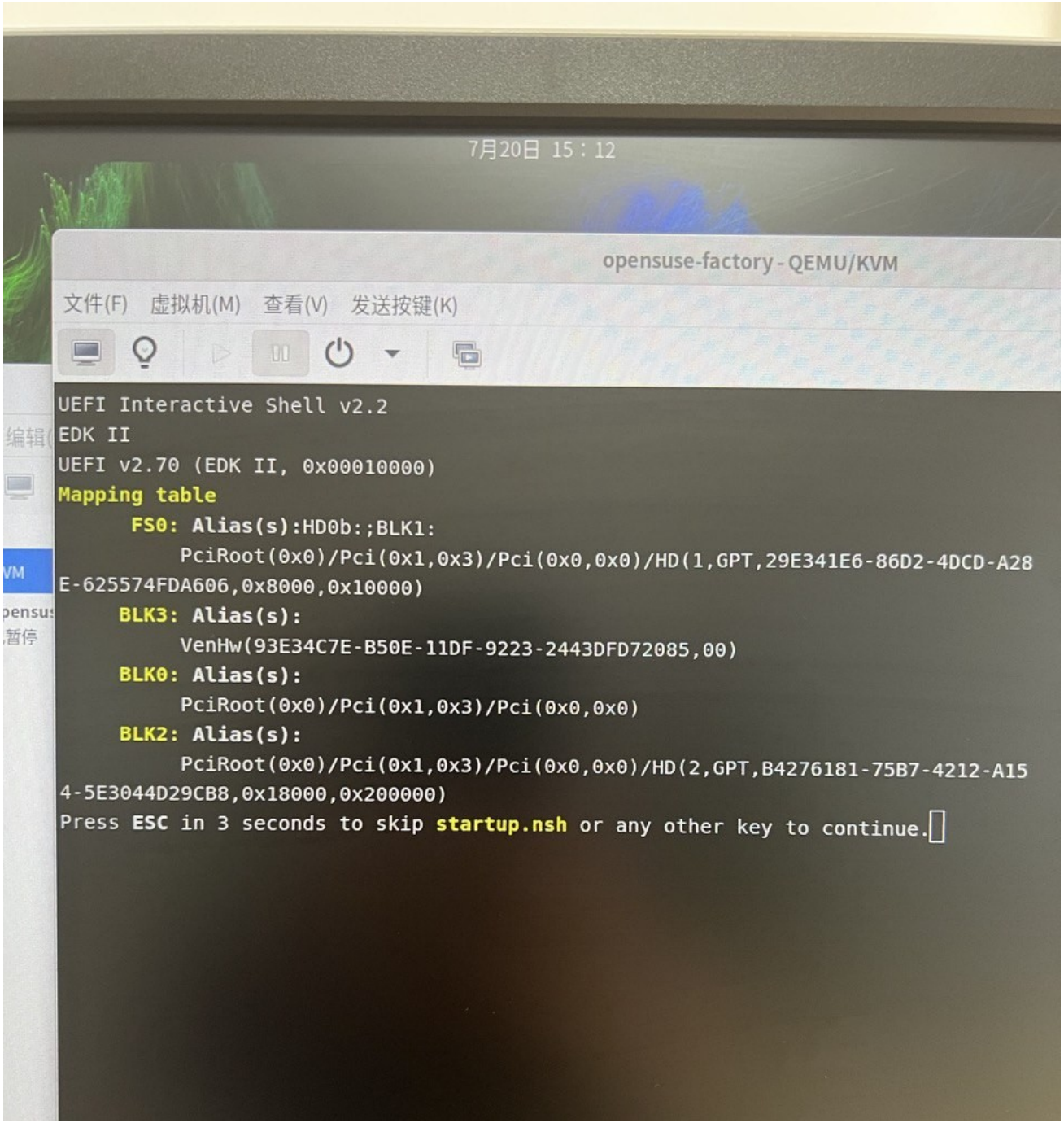
如果服务未激活，请手动激活并启动服务：

systemctl enable libvirtd

systemctl start libvirtd

systemctl status libvirtd

5.3 EFI 启动失败


7月20日 15：12
opensuse-factory - QEMU/KVM
文件(F) 虚拟机(M) 查看(V) 发送按键(K)
UEFI Interactive Shell v2.2
EDK II
UEFI v2.70 (EDK II, 0x00010000)
Mapping table
FS0: Alias(s):HD0b:;BLK1:
PciRoot(0x0)/Pci(0x1,0x3)/Pci(0x0,0x0)/HD(1,GPT,29E341E6-86D2-4DCD-A28E-625574FDA606,0x8000,0x10000)
BLK3: Alias(s):
VenHw(93E34C7E-B50E-11DF-9223-2443DFD72085,00)
BLK0: Alias(s):
PciRoot(0x0)/Pci(0x1,0x3)/Pci(0x0,0x0)
BLK2: Alias(s):
PciRoot(0x0)/Pci(0x1,0x3)/Pci(0x0,0x0)/HD(2,GPT,B4276181-75B7-4212-A154-5E3044D29CB8,0x18000,0x200000)
Press ESC in 3 seconds to skip startup.nsh or any other key to continue.

解决方法：删除虚拟机重建，多试几次，或者给虚拟机改个名

5.4 桥接模式下，虚拟机能 ping 通主机，主机也能 ping 通虚拟机，但虚机 ping 不通外网

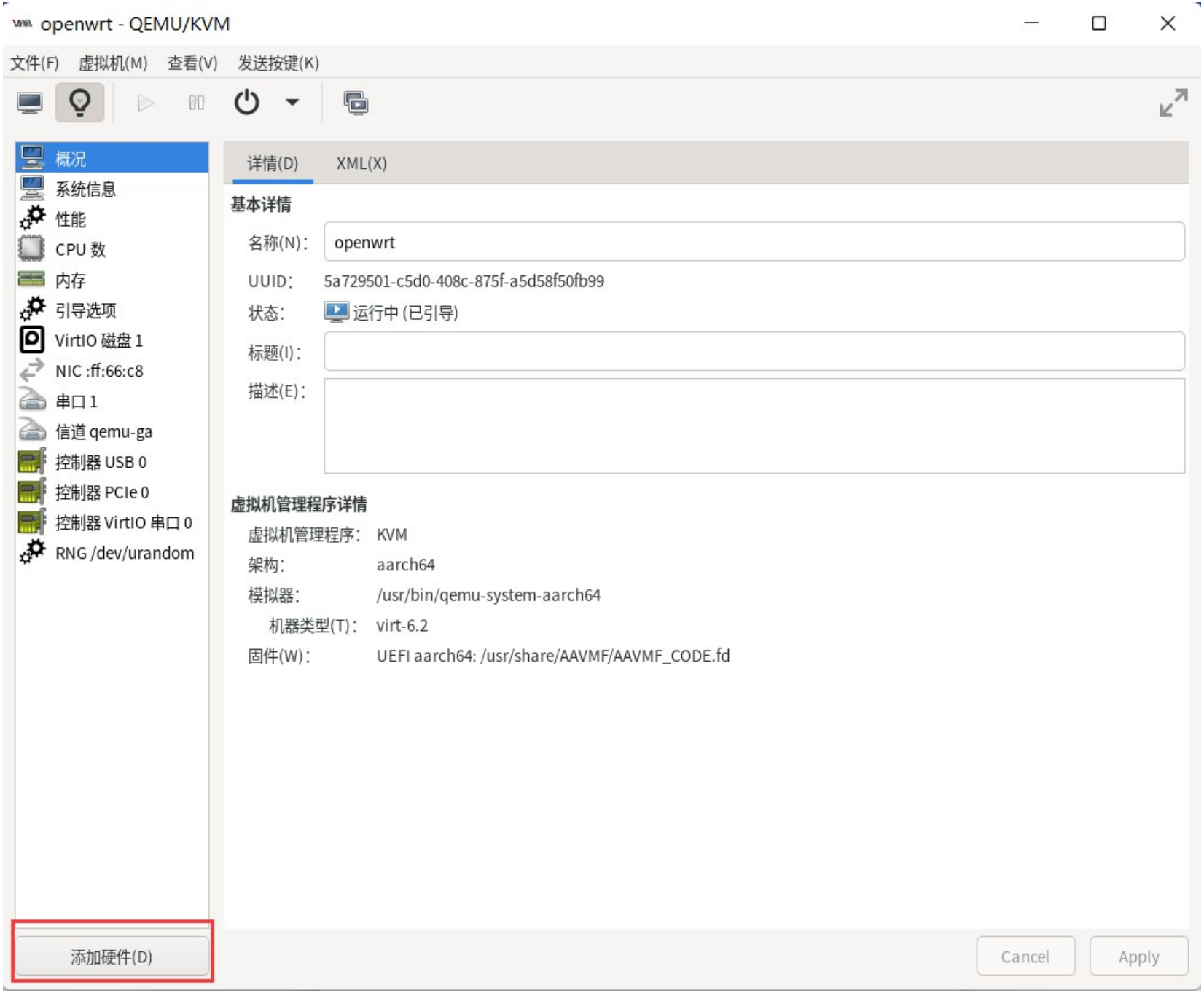
解决方法：一般是物理机防火墙开着引起的，可以关掉防火墙，或者在物理机的 /etc/sysctl.conf 里添加以下内容：
net.bridge.bridge-nf-call-ip6tables = 0
net.bridge.bridge-nf-call-iptables = 0
net.bridge.bridge-nf-call-arptables = 0

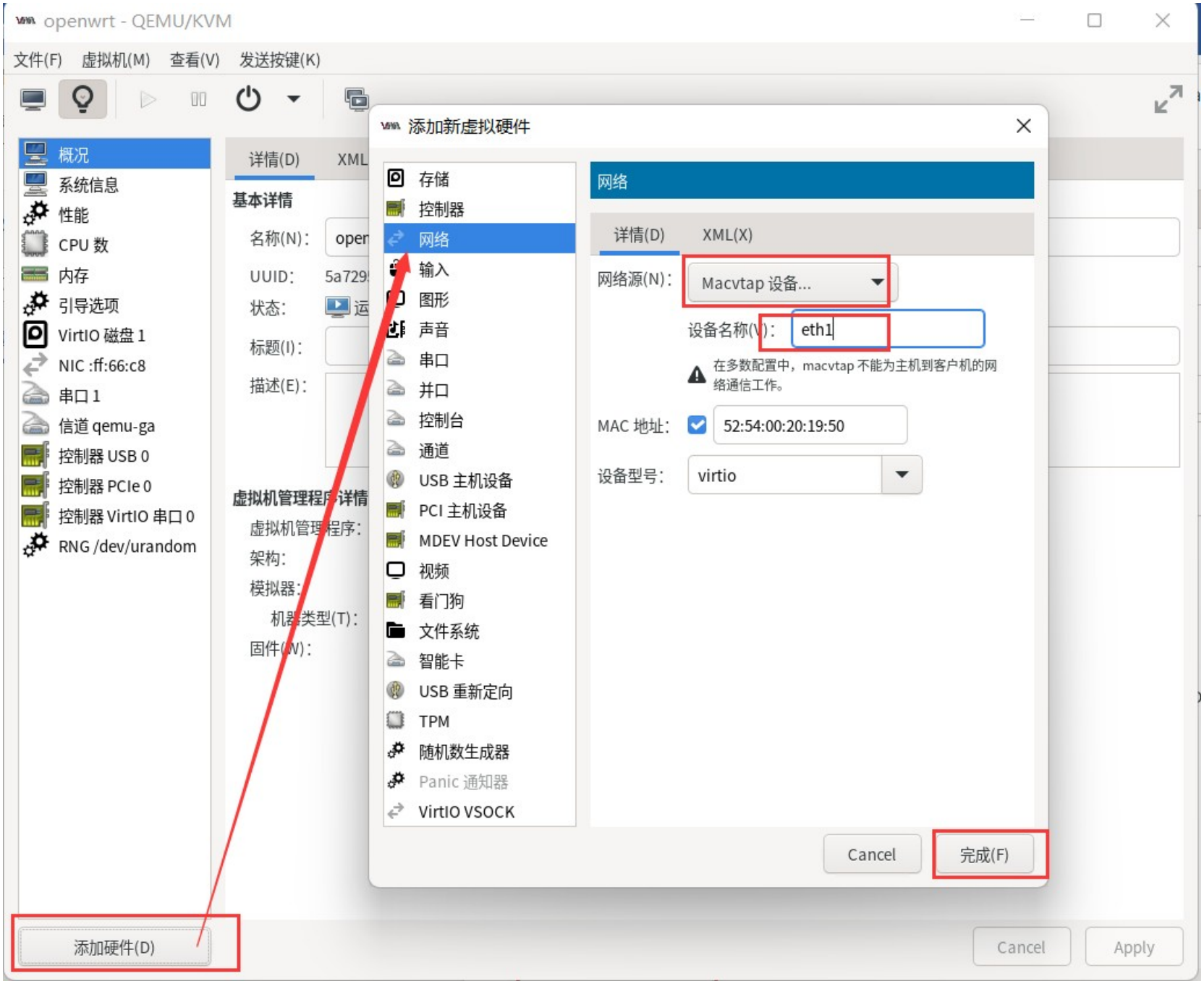
然后运行 sysctl -p

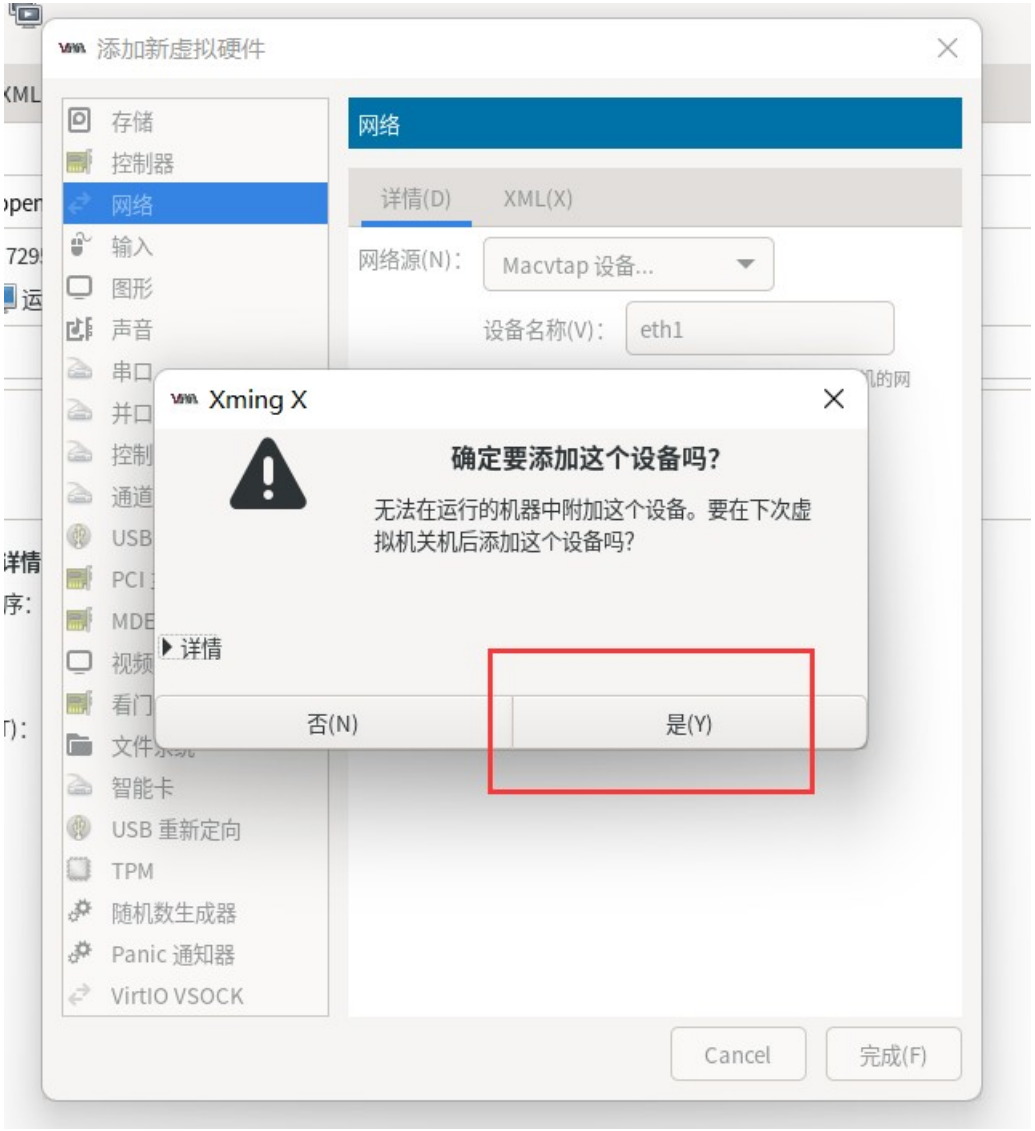
六、进阶用法

6.1 给虚拟机添加第 2 张网卡

前提是物理机有多余的网卡可用，无论是 usb 扩展的还是 pcie 扩展的都行，假设物理机的第二张网卡是 eth1，那么：


openwrt - QEMU/KVM
文件(F)
虚拟机(M)
查看(V)
发送按键(K)
概况
系统信息
性能
CPU 数
内存
引导选项
VirtIO 磁盘 1
NIC :ff:66:c8
串口 1
信道 qemu-ga
控制器 USB 0
控制器 PCIe 0
控制器 VirtIO 串口 0
RNG /dev/urandom
添加硬件(D)
详情(D)
XML(X)
基本详情
名称(N):
openwrt
UUID:
5a729501-c5d0-408c-875f-a5d58f50fb99
状态:
运行中 (已引导)
标题(I):
描述(E):
虚拟机管理程序详情
虚拟机管理程序:
KVM
架构:
aarch64
模拟器:
/usr/bin/qemu-system-aarch64
机器类型(T):
virt-6.2
固件(W):
UEFI aarch64: /usr/share/AAVMF/AAVMF_CODE.fd
Cancel
Apply

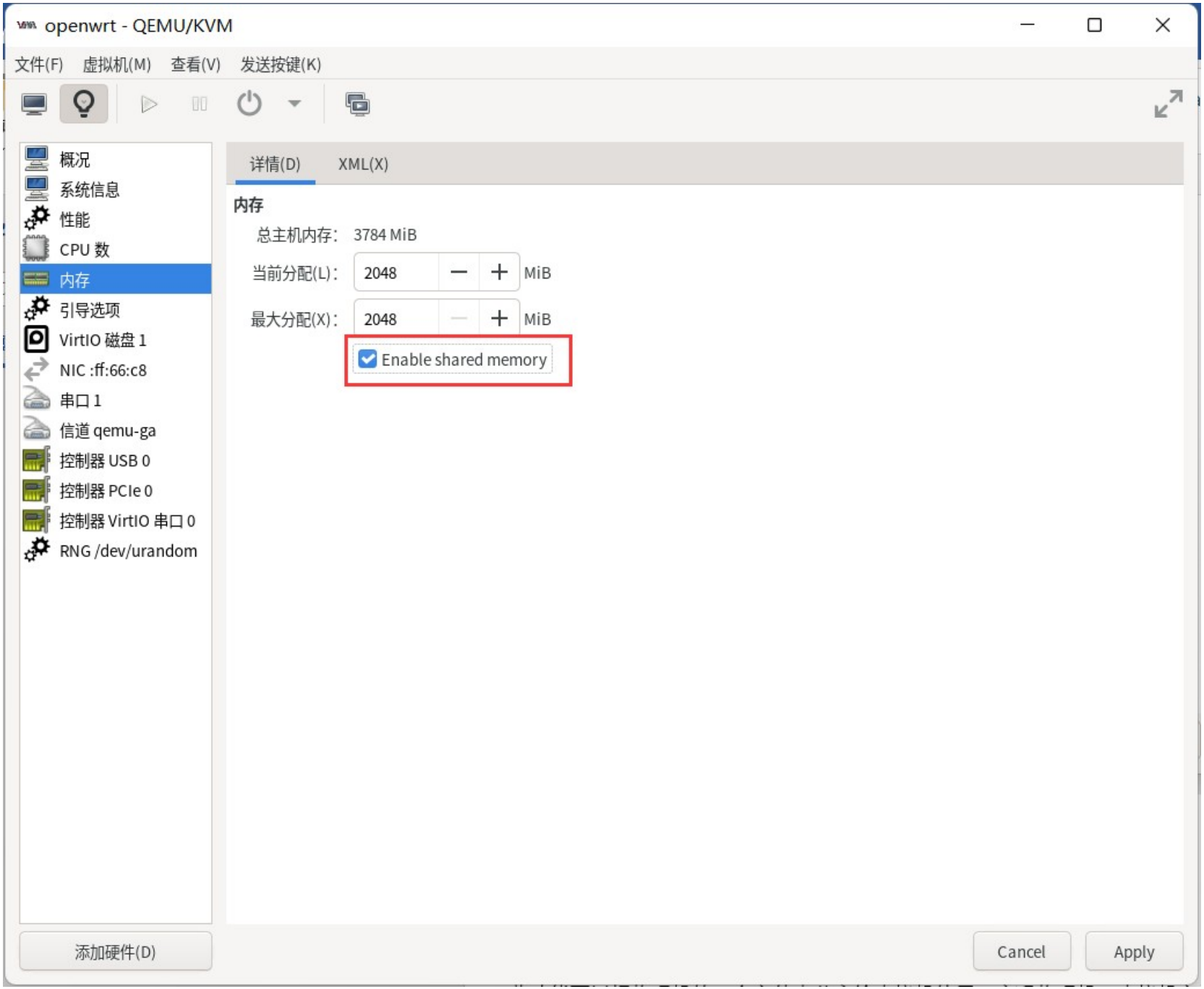

openwrt - QEMU/KVM
文件(F)
虚拟机(M)
查看(V)
发送按键(K)
概况
系统信息
性能
CPU 数
内存
引导选项
VirtIO 磁盘 1
NIC :ff:66:c8
串口 1
信道 qemu-ga
控制器 USB 0
控制器 PCIe 0
控制器 VirtIO 串口 0
RNG /dev/urandom
详情(D)
XML
基本详情
名称(N):
UUID:
状态:
标题(I):
描述(E):
虚拟机管理程序详情
虚拟机管理程序:
架构:
模拟器:
机器类型(T):
固件(W):
添加硬件(D)
Cancel
Apply
添加新虚拟硬件
存储
控制器
网络
输入
图形
声音
串口
并口
控制台
通道
USB 主机设备
PCI 主机设备
MDEV Host Device
视频
看门狗
文件系统
智能卡
USB 重新定向
TPM
随机数生成器
Panic 通知器
VirtIO VSOCK
网络
详情(D)
XML(X)
网络源(N):
Macvtap 设备...
设备名称(V):
eth1
在多数配置中，macvtap 不能为主机到客户机的网络通信工作。
MAC 地址:
52:54:00:20:19:50
设备型号:
virtio
Cancel
完成(F)



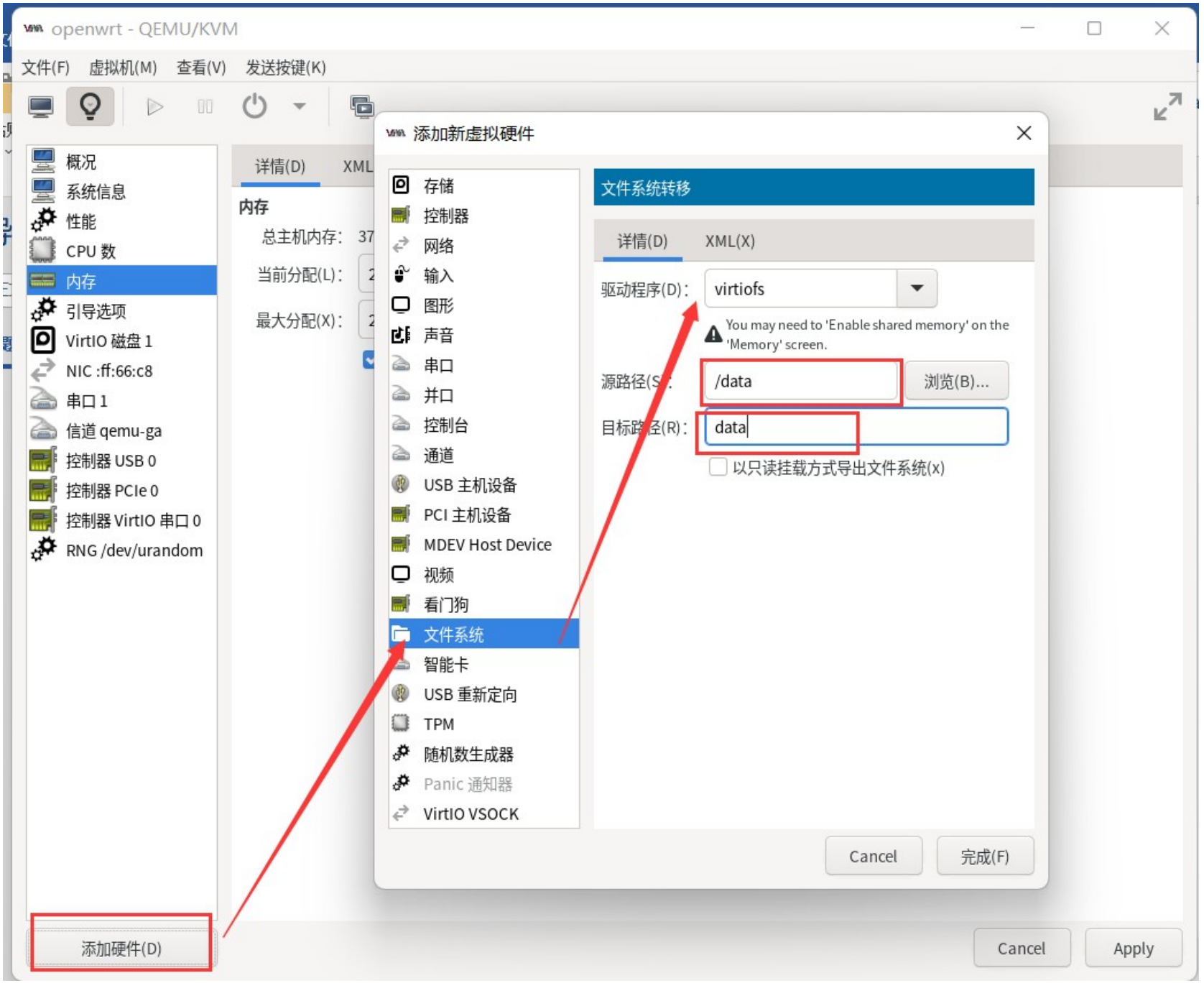
新添的网卡要关闭虚拟机之后才会出现，下次启动生效。

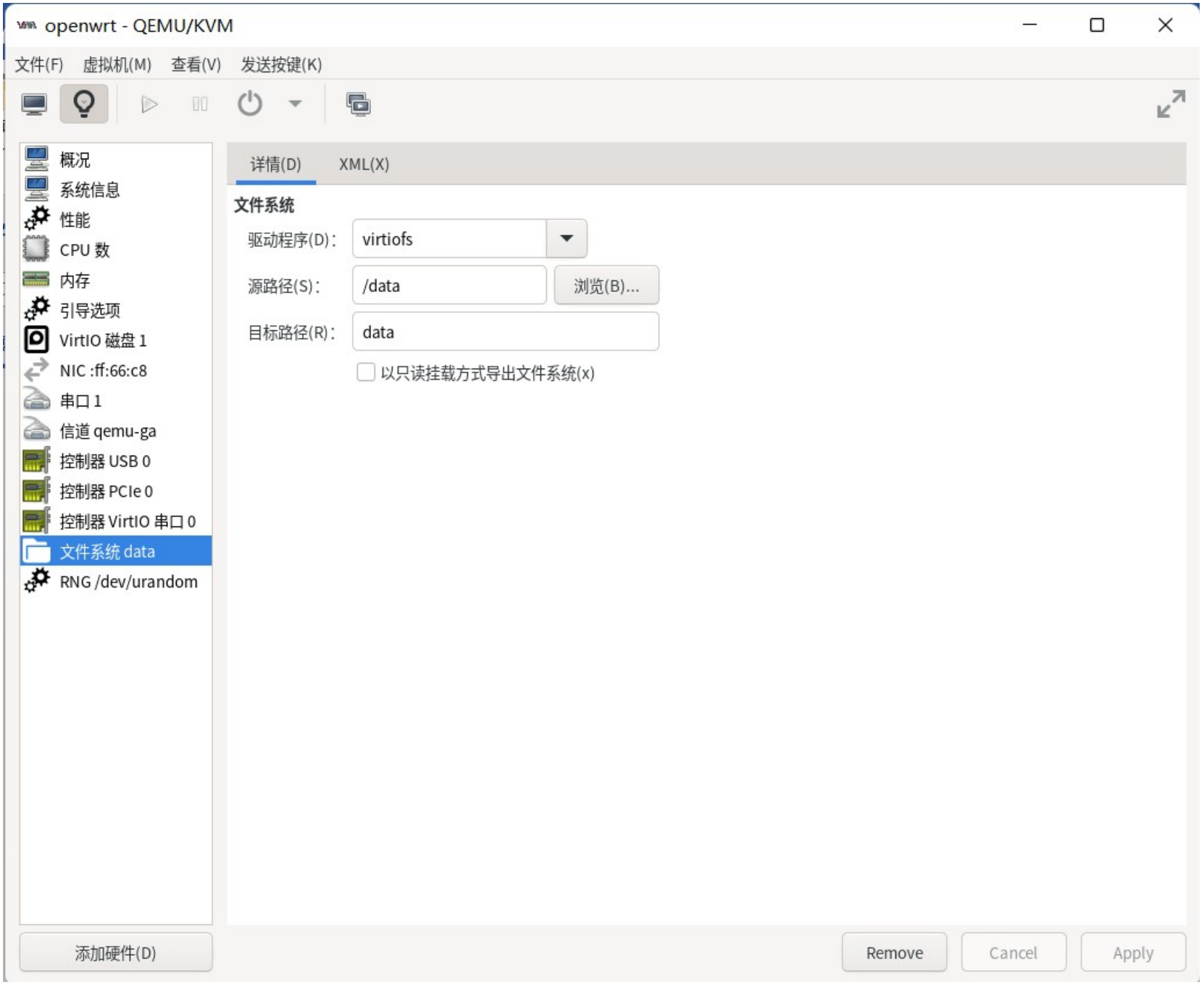
6.2 给虚拟机添加共享文件系统

此功能可以把物理机的一个文件夹共享给虚拟机使用，实现物理机、虚拟机之间文件共享，或多个虚拟机之间文件共享，非常实用!
首先要关闭虚拟机，把内存的 shared memory 选项打开：


openwrt - QEMU/KVM
文件(F)
虚拟机(M)
查看(V)
发送按键(K)
概况
系统信息
性能
CPU 数
内存
引导选项
VirtIO 磁盘 1
NIC :ff:66:c8
串口 1
信道 qemu-ga
控制器 USB 0
控制器 PCIe 0
控制器 VirtIO 串口 0
RNG /dev/urandom
详情(D)
XML(X)
内存
总主机内存: 3784 MiB
当前分配(L): 2048 MiB
最大分配(X): 2048 MiB
Enable shared memory
添加硬件(D)
Cancel
Apply

然后添加硬件，选择“文件系统”, 驱动程序选择“virtiofs”, 源路径选择物理机上已存在的某个文件夹，目标路径随便编个名字（例如 data）


openwrt - QEMU/KVM
文件(F)
虚拟机(M)
查看(V)
发送按键(K)
概况
系统信息
性能
CPU 数
内存
引导选项
VirtIO 磁盘 1
NIC :ff:66:c8
串口 1
信道 qemu-ga
控制器 USB 0
控制器 PCIe 0
控制器 VirtIO 串口 0
RNG /dev/urandom
添加硬件(D)
详情(D)
XML
内存
总主机内存:
当前分配(L):
最大分配(X):
Cancel
Apply
添加新虚拟硬件
存储
控制器
网络
输入
图形
声音
串口
并口
控制台
通道
USB 主机设备
PCI 主机设备
MDEV Host Device
视频
看门狗
文件系统
智能卡
USB 重新定向
TPM
随机数生成器
Panic 通知器
VirtIO VSOCK
文件系统转移
详情(D)
XML(X)
驱动程序(D):
virtiofs
You may need to 'Enable shared memory' on the 'Memory' screen.
源路径(S):
/data
浏览(B)...
目标路径(R):
data
以只读挂载方式导出文件系统(x)
Cancel
完成(F)


openwrt - QEMU/KVM
文件(F)
虚拟机(M)
查看(V)
发送按键(K)
概况
系统信息
性能
CPU 数
内存
引导选项
VirtIO 磁盘 1
NIC :ff:66:c8
串口 1
信道 qemu-ga
控制器 USB 0
控制器 PCIe 0
控制器 VirtIO 串口 0
文件系统 data
RNG /dev/urandom
详情(D)
XML(X)
文件系统
驱动程序(D):
virtiofs
源路径(S):
/data
浏览(B)...
目标路径(R):
data
以只读挂载方式导出文件系统(x)
添加硬件(D)
Remove
Cancel
Apply

然后启动虚拟机
在虚机中输入命令：
mkdir /mnt/data
mount -t virtiofs data /mnt/data
df -h

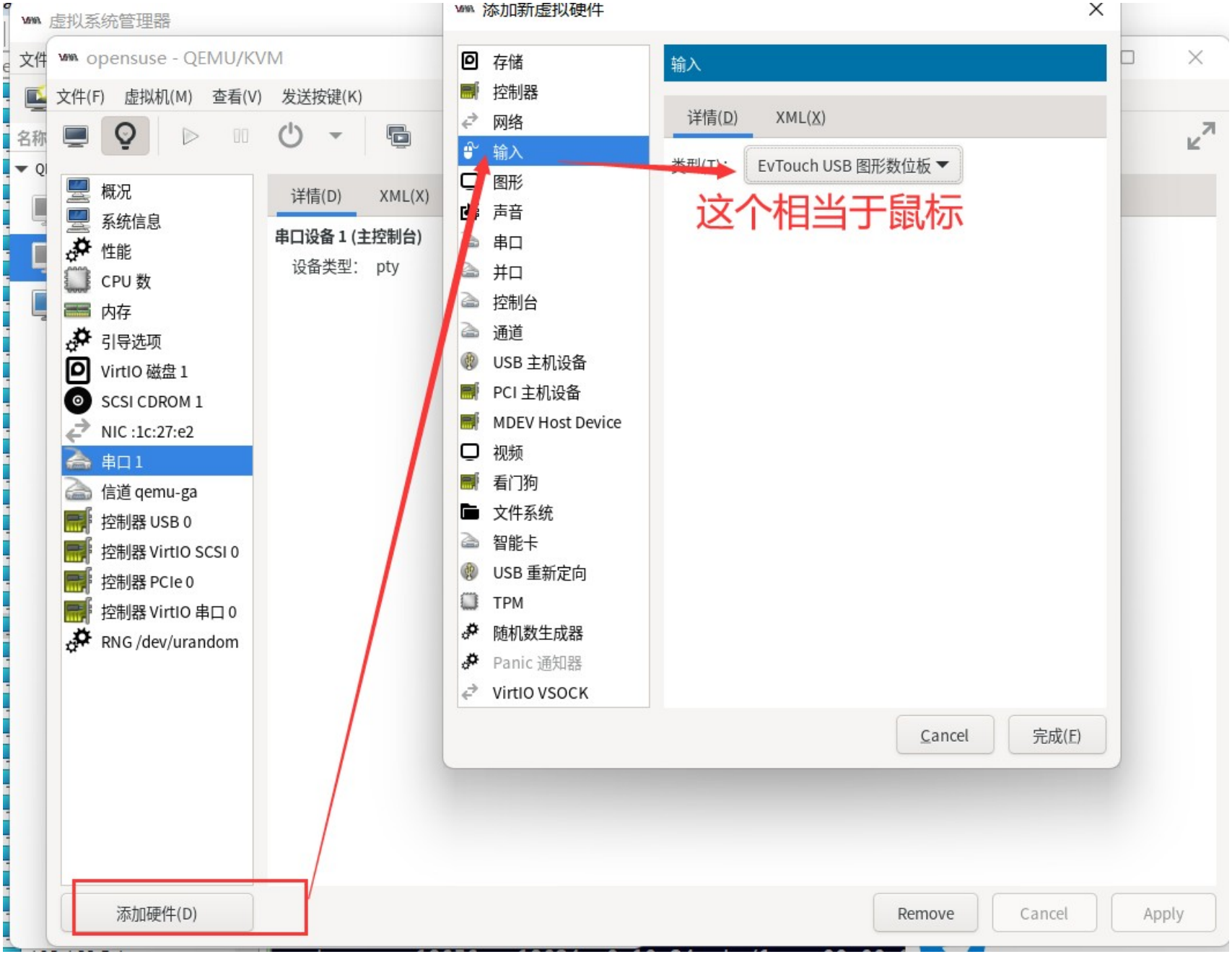
```
root@vm27:/# mkdir /mnt/data
mkdir: can't create directory '/mnt/data': File exists
root@vm27:/# mount -t virtiofs data /mnt/data
root@vm27:/# df -h
Filesystem               Size      Used Available Use% Mounted on
udev                   512.0K         0    512.0K   0% /dev
tmpfs                  198.1M     92.0K    198.0M   0% /run
/dev/vda2                1.0G    512.0M    409.4M  56% /
tmpfs                  990.6M     17.2M    973.5M   2% /tmp
tmpfs                  512.0K         0    512.0K   0% /dev
cgroup                 990.6M         0    990.6M   0% /sys/fs/cgroup
/dev/vda4               14.0G      4.0M     13.4G   0% /mnt/vda4
/dev/vda3                1.0G      3.8M    904.6M   0% /mnt/vda3
/dev/vda1               31.9M      1.4M     30.5M   4% /boot/efi
/dev/vda4               14.0G      4.0M     13.4G   0% /mnt/vda4/docker
/dev/vda4               14.0G      4.0M     13.4G   0% /mnt/vda4/docker/btrfs
data                   100.2G     89.6G     10.5G  89% /mnt/data
root@vm27:/#
```

挂载成功，如果想要开机自动挂载的话，可以把挂载命令添加到 /etc/rc.local 里

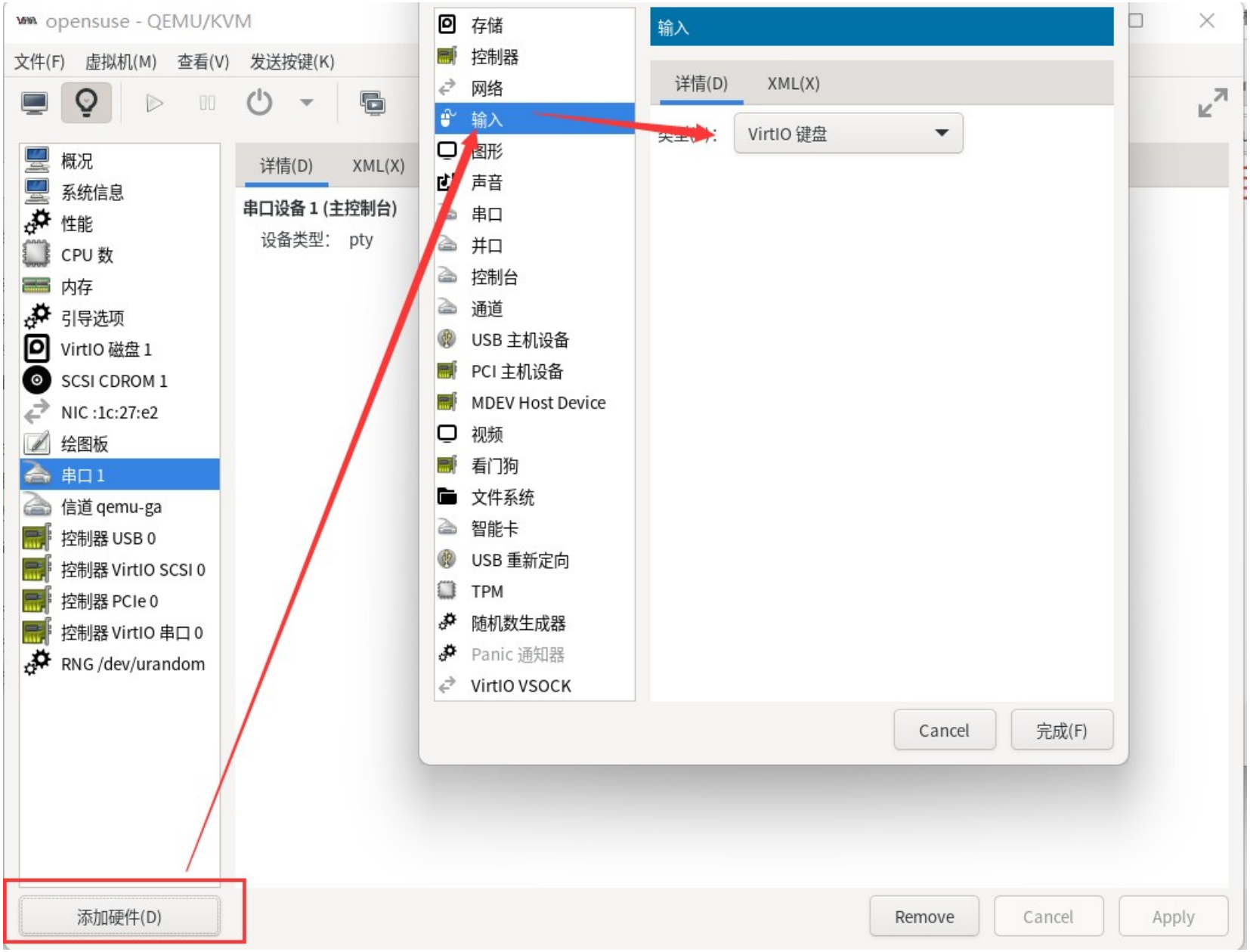
```
# Put your custom commands here that should be executed once
# the system init finished. By default this file does nothing.
mount -t virtiofs data  /mnt/data
exit 0
~
```

6.3 给虚拟机添加显卡（openwrt 没啥用）

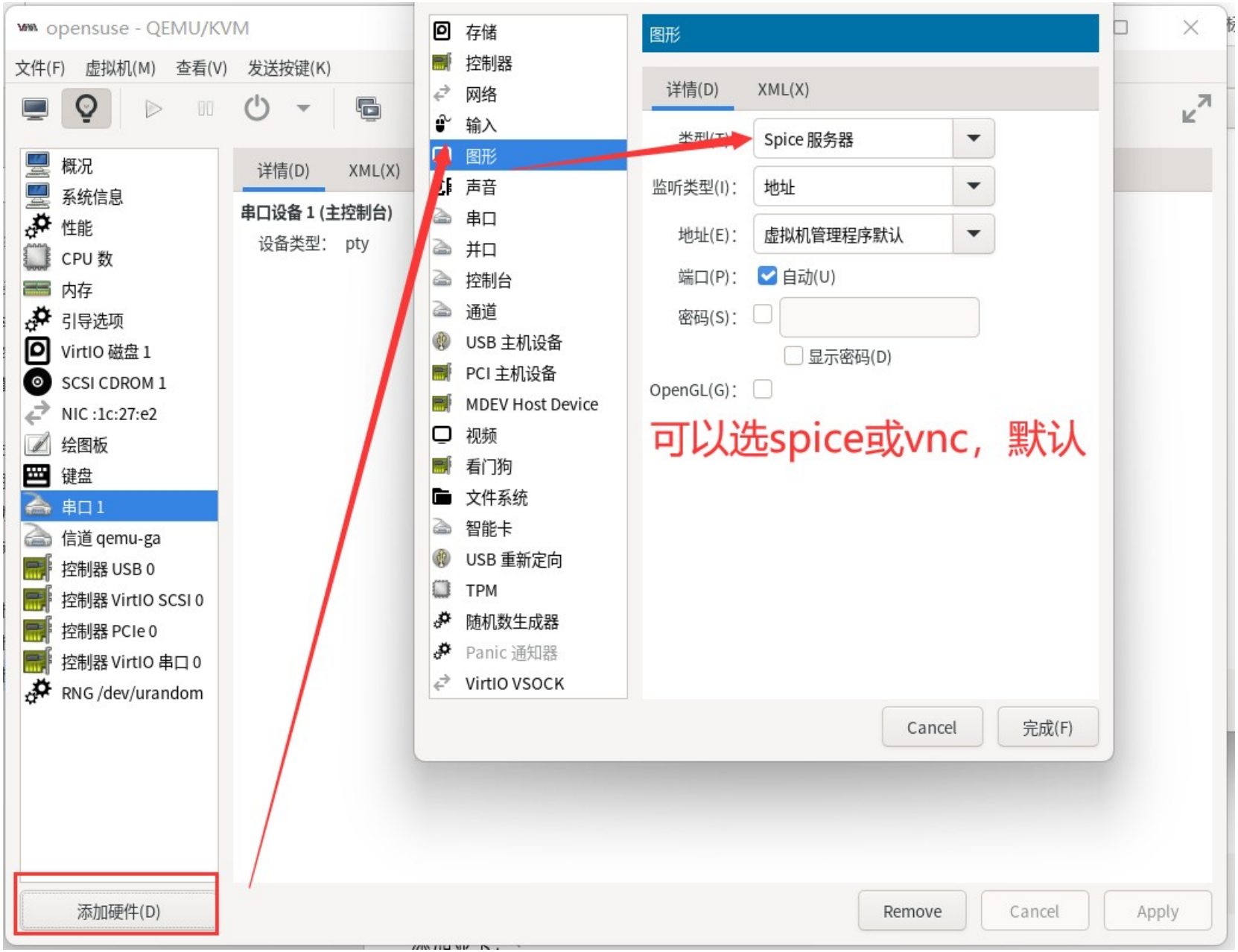
虽然对于 openwrt 没用，但对于其它 linux 发行版有用，如果想在 armbian 里运行另一个 linux（debian,ubuntu,openSUSE,archlinux,centos,gentoo,国产麒麟，国产统信 uos 等等），那这一步是必需的：
添加鼠标：


虚拟系统管理器
opensuse - QEMU/KVM
文件(F) 虚拟机(M) 查看(V) 发送按键(K)
概况
系统信息
性能
CPU 数
内存
引导选项
VirtIO 磁盘 1
SCSI CDROM 1
NIC :1c:27:e2
串口 1
信道 qemu-ga
控制器 USB 0
控制器 VirtIO SCSI 0
控制器 PCIe 0
控制器 VirtIO 串口 0
RNG /dev/urandom
详情(D) XML(X)
串口设备 1 (主控制台)
设备类型: pty
添加硬件(D)
Remove
Cancel
Apply
添加新虚拟硬件
存储
控制器
网络
输入
图形
声音
串口
并口
控制台
通道
USB 主机设备
PCI 主机设备
MDEV Host Device
视频
看门狗
文件系统
智能卡
USB 重新定向
TPM
随机数生成器
Panic 通知器
VirtIO VSOCK
输入
详情(D) XML(X)
EvTouch USB 图形数位板
这个相当于鼠标
Cancel
完成(F)

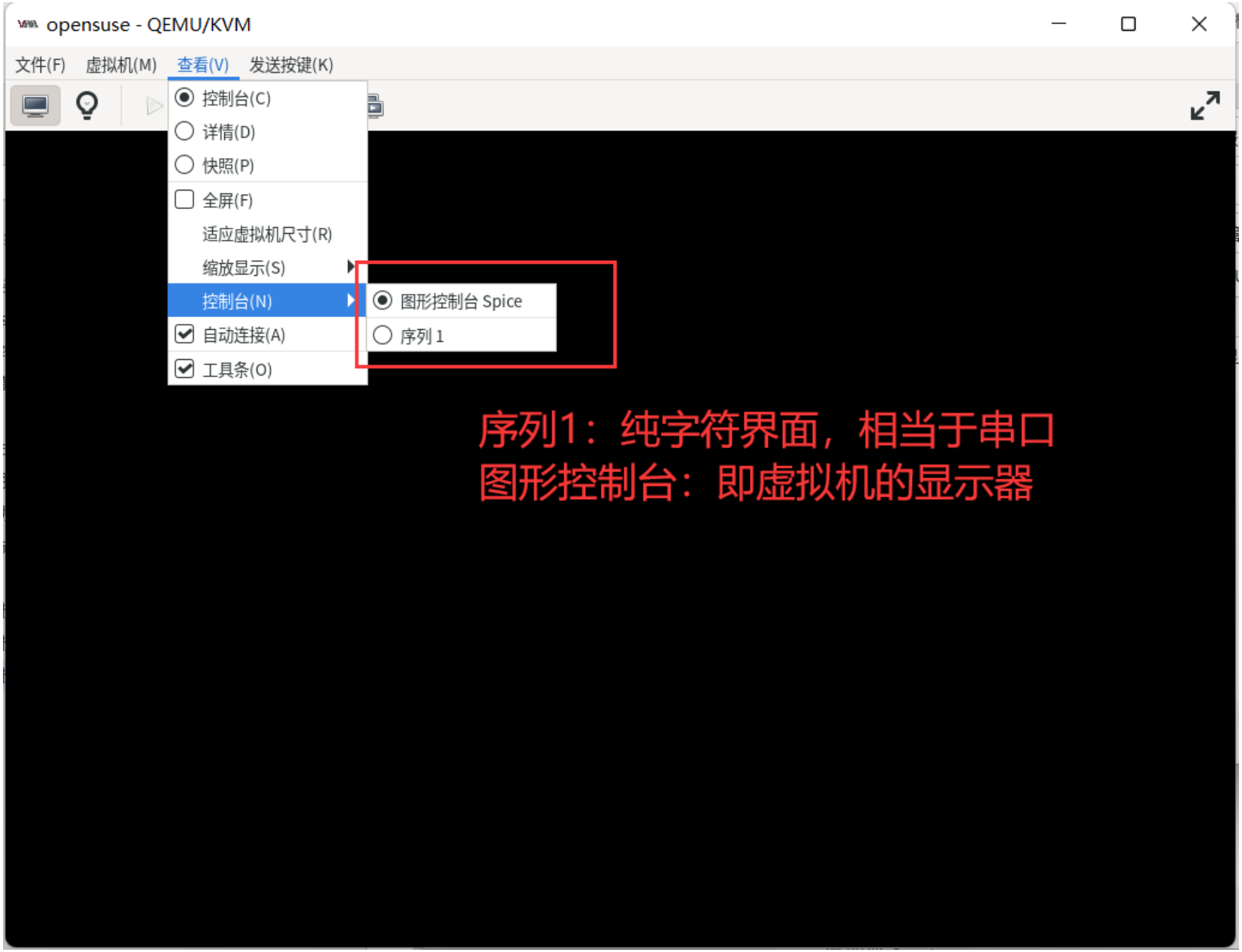
添加键盘:


opensuse - QEMU/KVM
文件(F)
虚拟机(M)
查看(V)
发送按键(K)
概况
系统信息
性能
CPU 数
内存
引导选项
VirtIO 磁盘 1
SCSI CDROM 1
NIC :1c:27:e2
绘图板
串口 1
信道 qemu-ga
控制器 USB 0
控制器 VirtIO SCSI 0
控制器 PCIe 0
控制器 VirtIO 串口 0
RNG /dev/urandom
详情(D)
XML(X)
串口设备 1 (主控制台)
设备类型: pty
添加硬件(D)
Remove
Cancel
Apply
存储
控制器
网络
输入
图形
声音
串口
并口
控制台
通道
USB 主机设备
PCI 主机设备
MDEV Host Device
视频
看门狗
文件系统
智能卡
USB 重新定向
TPM
随机数生成器
Panic 通知器
VirtIO VSOCK
输入
详情(D)
XML(X)
VirtIO 键盘
Cancel
完成(F)

添加显卡:

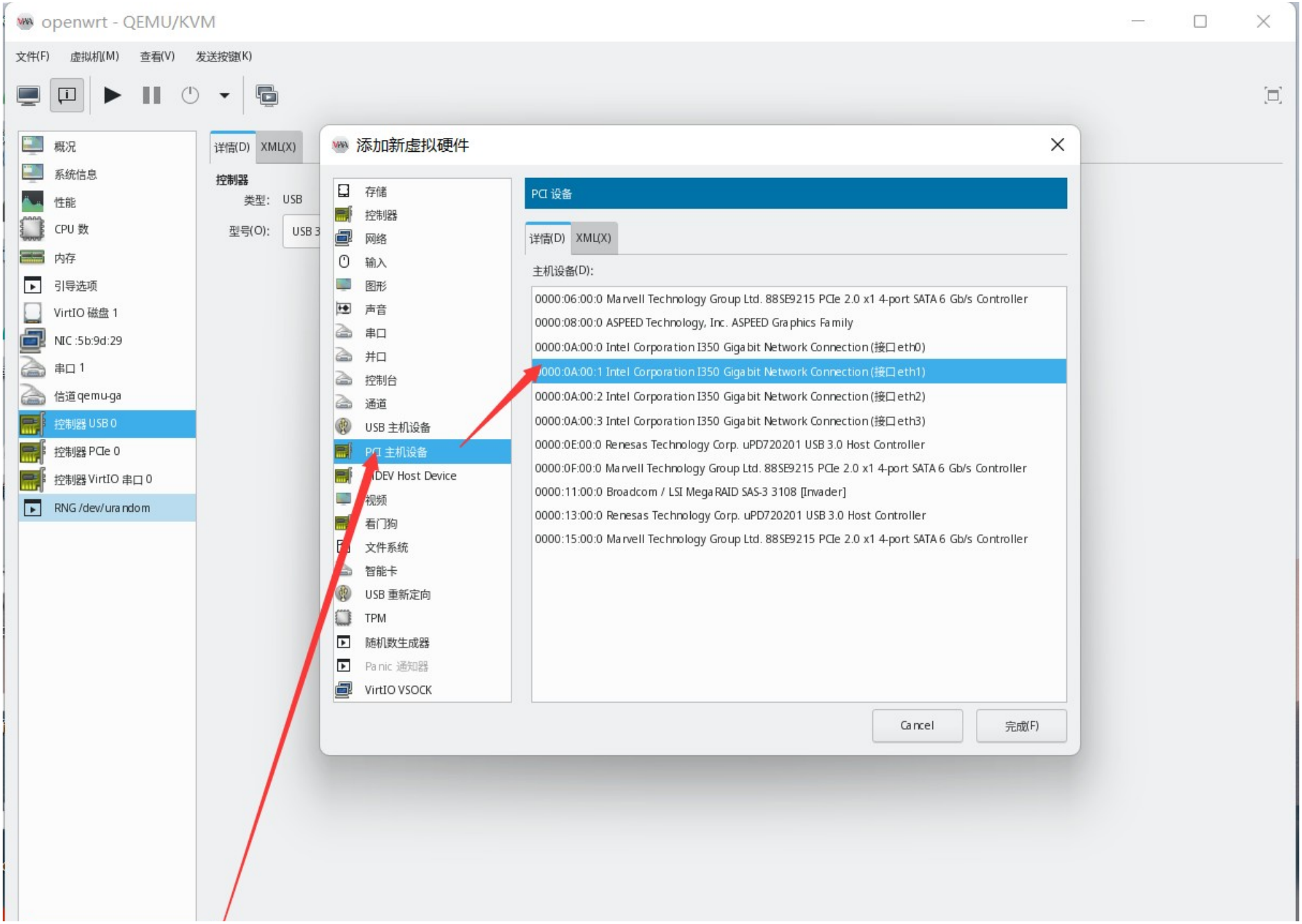

opensuse - QEMU/KVM
文件(F) 虚拟机(M) 查看(V) 发送按键(K)
概况
系统信息
性能
CPU 数
内存
引导选项
VirtIO 磁盘 1
SCSI CDROM 1
NIC :1c:27:e2
绘图板
键盘
串口 1
信道 qemu-ga
控制器 USB 0
控制器 VirtIO SCSI 0
控制器 PCIe 0
控制器 VirtIO 串口 0
RNG /dev/urandom
添加硬件(D)
详情(D) XML(X)
串口设备 1 (主控制台)
设备类型: pty
存储
控制器
网络
输入
图形
声音
串口
并口
控制台
通道
USB 主机设备
PCI 主机设备
MDEV Host Device
视频
看门狗
文件系统
智能卡
USB 重新定向
TPM
随机数生成器
Panic 通知器
VirtIO VSOCK
图形
详情(D) XML(X)
Spice 服务器
监听类型(I): 地址
地址(E): 虚拟机管理程序默认
端口(P): 自动(U)
密码(S):
显示密码(D)
OpenGL(G):
可以选spice或vnc，默认
Cancel
完成(F)
Remove
Cancel
Apply

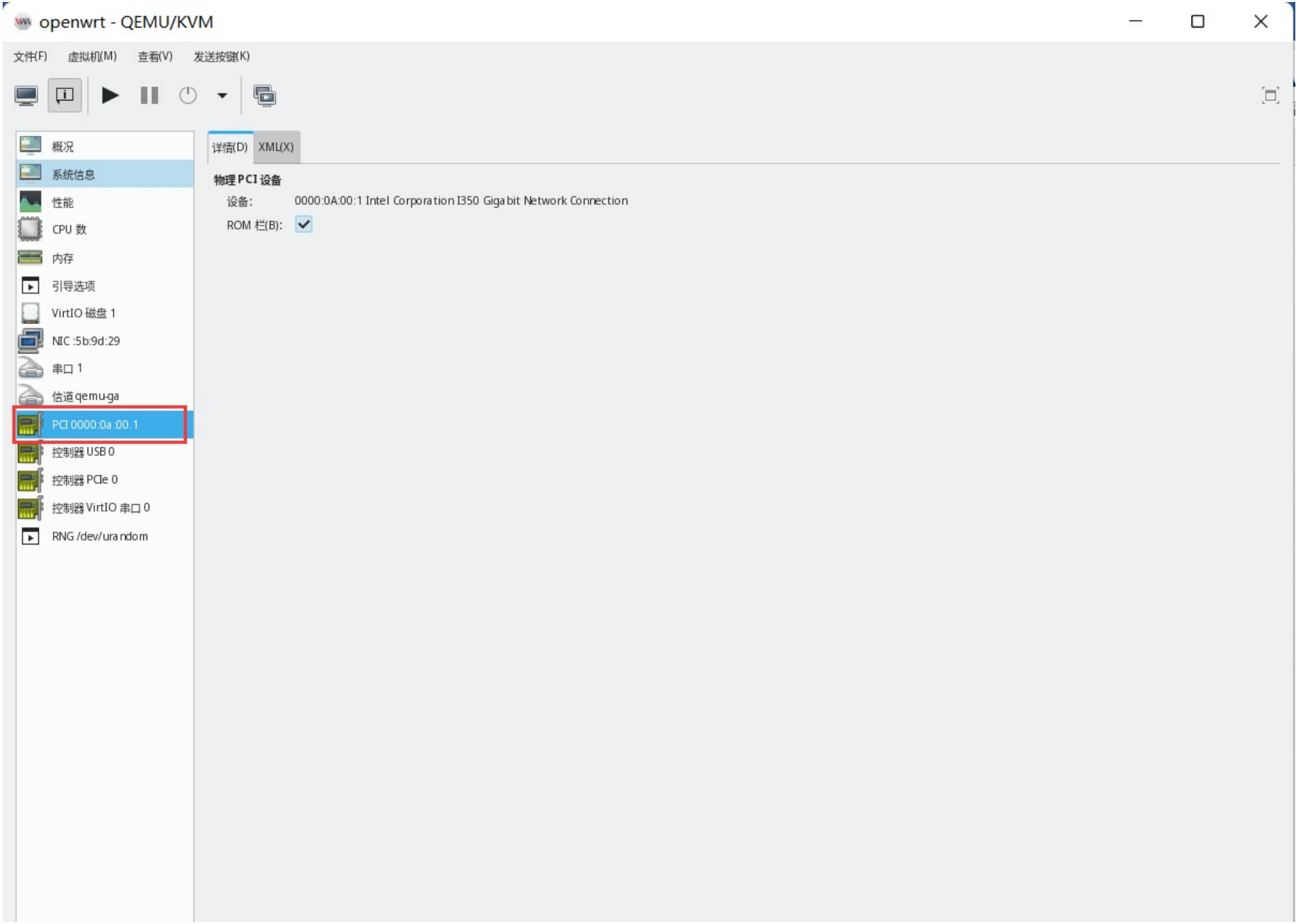
启动虚拟机：

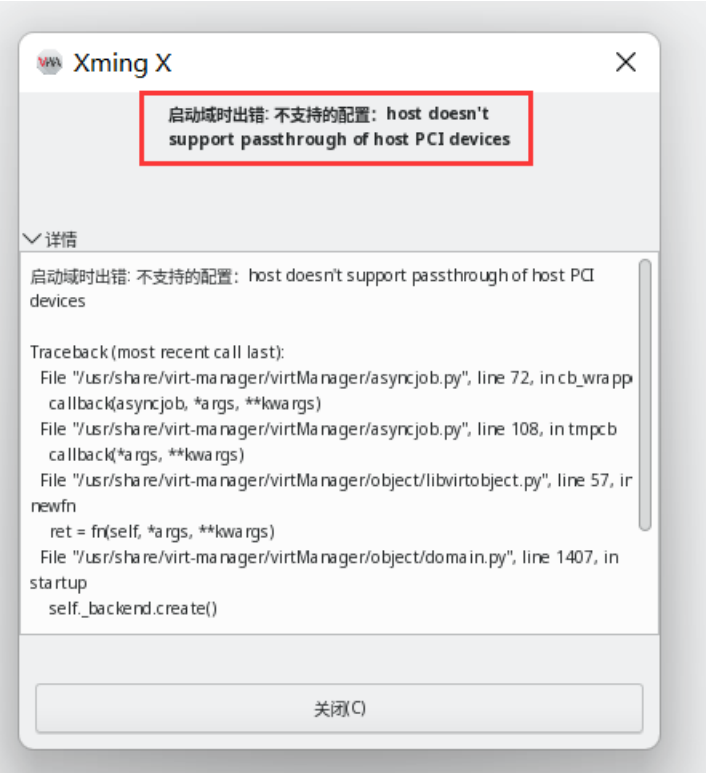

opensuse - QEMU/KVM
文件(F)
虚拟机(M)
查看(V)
发送按键(K)
控制台(C)
详情(D)
快照(P)
全屏(F)
适应虚拟机尺寸(R)
缩放显示(S)
控制台(N)
自动连接(A)
工具条(O)
图形控制台 Spice
序列 1
序列1：纯字符界面，相当于串口
图形控制台：即虚拟机的显示器

6.4 给虚拟机添加直通设备

这需要物理机支持 iommu，一般的电视盒子就别想了，目前即使正规的 arm64 服务器也很少支持。


openwrt - QEMU/KVM
文件(F)
虚拟机(M)
查看(V)
发送按键(K)
概况
系统信息
性能
CPU 数
内存
引导选项
VirtIO 磁盘 1
NIC :5b:9d:29
串口 1
信道 qemu-ga
控制器 USB 0
控制器 PCIe 0
控制器 VirtIO 串口 0
RNG /dev/urandom
详情(D)
XML(X)
控制器
类型: USB
型号(O):
添加新虚拟硬件
存储
控制器
网络
输入
图形
声音
串口
并口
控制台
通道
USB 主机设备
PCI 主机设备
IDEV Host Device
视频
看门狗
文件系统
智能卡
USB 重新定向
TPM
随机数生成器
Panic 通知器
VirtIO VSOCK
PCI 设备
详情(D)
XML(X)
主机设备(D):
0000:06:00:0 Marvell Technology Group Ltd. 88SE9215 PCIe 2.0 x1 4-port SATA 6 Gb/s Controller
0000:08:00:0 ASPEED Technology, Inc. ASPEED Graphics Family
0000:0A:00:0 Intel Corporation I350 Gigabit Network Connection (接口 eth0)
0000:0A:00:1 Intel Corporation I350 Gigabit Network Connection (接口 eth1)
0000:0A:00:2 Intel Corporation I350 Gigabit Network Connection (接口 eth2)
0000:0A:00:3 Intel Corporation I350 Gigabit Network Connection (接口 eth3)
0000:0E:00:0 Renesas Technology Corp. uPD720201 USB 3.0 Host Controller
0000:0F:00:0 Marvell Technology Group Ltd. 88SE9215 PCIe 2.0 x1 4-port SATA 6 Gb/s Controller
0000:11:00:0 Broadcom / LSI MegaRAID SAS-3 3108 [Invader]
0000:13:00:0 Renesas Technology Corp. uPD720201 USB 3.0 Host Controller
0000:15:00:0 Marvell Technology Group Ltd. 88SE9215 PCIe 2.0 x1 4-port SATA 6 Gb/s Controller
Cancel
完成(F)


openwrt - QEMU/KVM
文件(F)
虚拟机(M)
查看(V)
发送按键(K)
概况
系统信息
性能
CPU 数
内存
引导选项
VirtIO 磁盘 1
NIC :5b:9d:29
串口 1
信道 qemu-ga
PCI 0000:0a:00.1
控制器 USB 0
控制器 PCIe 0
控制器 VirtIO 串口 0
RNG /dev/urandom
详情(D)
XML(X)
物理 PCI 设备
设备:
0000:0A:00:1 Intel Corporation I350 Gigabit Network Connection
ROM 栏(B):



出现这个提示就是不支持直通 ☹

七、固件升级

每次固件发布会有 2 个文件：
openwrt_qemu-aarch64_generic_vm_k5.18.13-flippy-75+.img
openwrt_qemu-aarch64_generic_vm_k5.18.13-flippy-75+.qcow2
其中，后缀为.qcow2 的文件是首次创建虚拟机用的，而另一个后缀为.img 的文件就用于升级的。

7.1 命令行升级方法：

1. 把 openwrt_qemu-aarch64_generic_vm_k5.18.13-flippy-75+.img 及附带的升级脚本上传至虚拟机的 /mnt/vda4 目录下
   （7z 压缩包里也会同时包含一个升级脚本：update-kvm-openwrt.sh，与/usr/sbin/openwrt-update-kvm 是同一个文件，但版本可能更新一些）
2. cd /mnt/vda4
3. /usr/sbin/openwrt-update-kvm openwrt_qemu-aarch64_generic_vm_k5.18.13-flippy-75+.img
   或
   ./update-kvm-openwrt.sh openwrt_qemu-aarch64_generic_vm_k5.18.13-flippy-75+.img

7.2. 用“虚拟宝盒”应用进行升级

使用方法基本与“晶晨宝盒”相同

八、内核升级

内核升级即：只升级 kernel，不升级 openwrt 的应用。

8.1. 命令行升级

把 boot-xxxx.tar.gz、modules-xxxx.tar.gz 两个内核压缩包上传至 /mnt/vda4, 然后运行：
openwrt-kernel-kvm

8.2. 用“虚拟宝盒”应用进行升级

使用方法基本与“晶晨宝盒”相同