

Guest 网络设置

2014/4/10

renyl

1 Bridge 的设置

1.1 确认能 ping 通对联机

首先本机的一块物理网卡（以 eth2 为例）和对联机已经连接，确认和对联机能够 ping 通，假设对联机 IP 地址：192.168.2.2

```
# vim /etc/sysconfig/network-scripts/ ifcfg-eth2
DEVICE="eth2"
HWADDR="00:23:26:F1:29:A4"
ONBOOT="yes"
BOOTPROTO="static"
IPADDR=192.168.2.1
# ifdown eth2;ifup eth2
# ping 192.168.2.2
```

1.2 添加 br0

/etc/sysconfig/network-scripts/目录下添加 ifcfg-br0 文件，内容如下：

```
# vim /etc/sysconfig/network-scripts/ ifcfg-br0
DEVICE=br0
TYPE=Bridge    ※1
BOOTPROTO=none
ONBOOT=yes
```

※1: TYPE=Bridge 注意大小写敏感。

1.3 将 eth2 关联到 br0

在 eth2 配置文件中添加 BRIDGE=br0

```
# vim /etc/sysconfig/network-scripts/ ifcfg-eth2
DEVICE="eth2"
HWADDR="00:23:26:F1:29:A4"
ONBOOT="yes"
BOOTPROTO="static"
BRIDGE=br0
```

1.4 重启 br0，并设置 STP 为 yes

```
# ifdown br0;ifup br0
# brctl show
      bridge name      bridge id                STP enabled    interfaces
      br0               8000.c80aa933505e         no             eth2
# brctl stp br0 yes
      bridge name      bridge id                STP enabled    interfaces
      br0               8000.c80aa933505e         yes            eth2
```

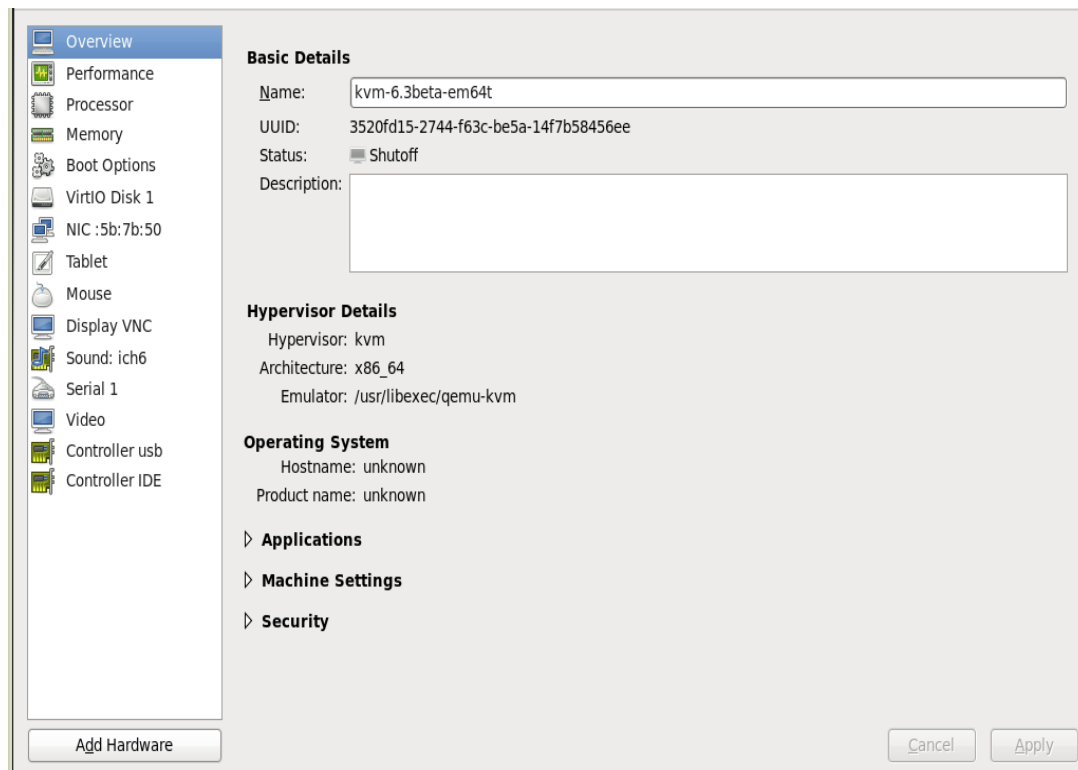
1.5 将 br0 添加到 guest 中

1.5.1 在 host 中执行 virt-manager 命令

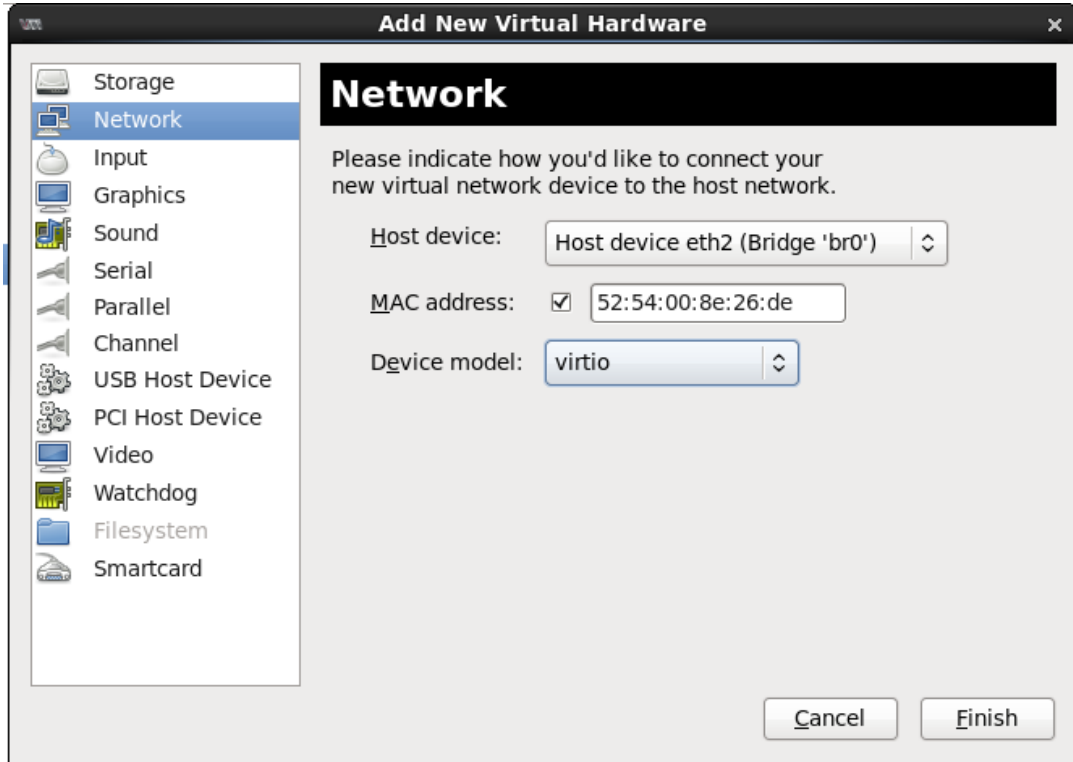
```
# virt-manager
```

1.5.2 为虚拟机添加虚拟网卡

点击 Add Hardware，添加虚拟网卡，如下图



1.5.2 选择 Network: Host device: br0, Device model: virtio



1.6 添加完虚拟网卡后，启动虚拟机

```
# virsh start 虚拟机名
```

1.7 在虚拟机中，设置 IP 地址

```
# vim /etc/sysconfig/network-scripts/ifcfg-eth1  
DEVICE="eth1"  
BOOTPROTO="static"  
HWADDR="52:54:00:8e:26:de"  
NM_CONTROLLED="yes"  
ONBOOT="yes"  
TYPE="Ethernet"  
IPADDR=192.168.2.1 ※3
```

※3 与对联机同一网段的 IP 地址

1.8 在虚拟机中，重启网络

```
# /etc/rc.d/init.d/network restart
# ifconfig | grep eth1
```

1.9 在虚拟机中，和对联机 ping 确认

```
# ping 对联机的 IP 地址
```

```
# ping 192.168.2.2
  PING 192.168.2.2 (192.168.2.2) 56(84) bytes of data.
  64 bytes from 192.168.2.2: icmp_seq=1 ttl=64 time=0.337 ms
  64 bytes from 192.168.2.2: icmp_seq=2 ttl=64 time=0.285 ms
```

2 PCI passthrough 的设定

2.1 host 的设定

2.1.1 在 grub.conf 的 kernel 行添加 intel_iommu=on

```
# vim /boot/efi/EFI/redhat/grub.conf
kernel /vmlinuz-2.6.32-262.el6.x86_64 ro
root=UUID=5abe1186-9917-41ec-b057-641f08b0eba5 5rd_NO_LUKS rd_NO_LVM
LANG=en_US.UTF-8 rd_NO_MD SYSFONT=latarcyrheb-sun16 crashkernel=128M
KEYBOARDTYPE=pc KEYTABLE=us rd_NO_DM rhgb quiet intel_iommu=on
```

2.1.2 重启系统

```
# reboot
```

2.1.3 BIOS 上有効性的确认

```
# dmesg | grep 'ACPI: DMAR'
ACPI: DMAR 0000000072bbb000 00048 (v01 FUJ PC 00000001 FJ 00000001)
```

2.1.4 IOMMU 驱动器有效性的确认

```
# dmesg | grep 'Virtualization Technology for Directed I/O'
PCI-DMA: Intel(R) Virtualization Technology for Directed I/O
```

2.1.5 IOMMU 驱动器有效性的确认

```
# lspci | grep Ethernet (网络) 或者 Fibre (光纤卡磁盘)
06:00.0 Ethernet controller: Intel Corporation 82576NS Gigabit Network Connection
(rev 01) ※4
```

※4 先頭の 06:00.0 为 PCI 番号, bus=0x06, slot=0x00, function=0x0。

2.1.6 从 virsh nodedev-list 指定特定设备

```
# virsh nodedev-list --tree | grep 06_00_0  
pci_0000_06_00_0    ※5
```

2.1.7 从 virsh dumpxml 确认设备是否是 FC 卡

```
# virsh nodedev-dumpxml pci_0000_06_00_0  
<device>  
  <name>pci_0000_06_00_0</name>  
  <parent>pci_0000_04_01_0</parent>  
  <driver>  
    <name>igb</name>  
  </driver>  
  <capability type='pci'>  
    <domain>0</domain>  
    <bus>6</bus>  
    <slot>0</slot>  
    <function>0</function>  
    <product id='0x150a'>82576NS Gigabit Network Connection</product>  
    <vendor id='0x8086'>Intel Corporation</vendor>  
    <capability type='virt_functions'>  
      </capability>  
    </capability>  
  </device>
```

2.1.8 在 Host 上做 detach 操作

```
# virsh nodedev-dettach pci_0000_06_00_0  
Device pci_0000_06_00_0 detached
```

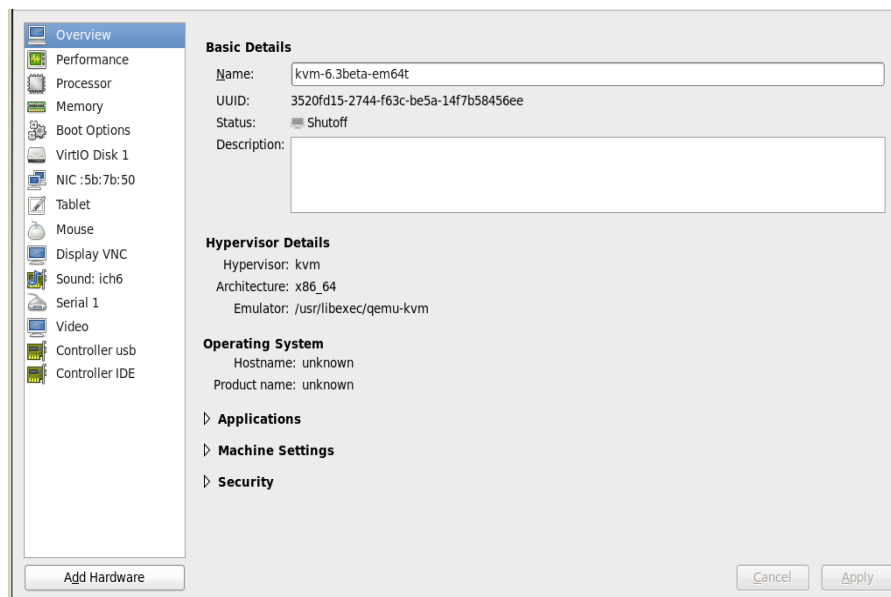
2.2 虚拟机添加 PCI 设备

2.2.1 在 host 中执行 virt-manager 命令

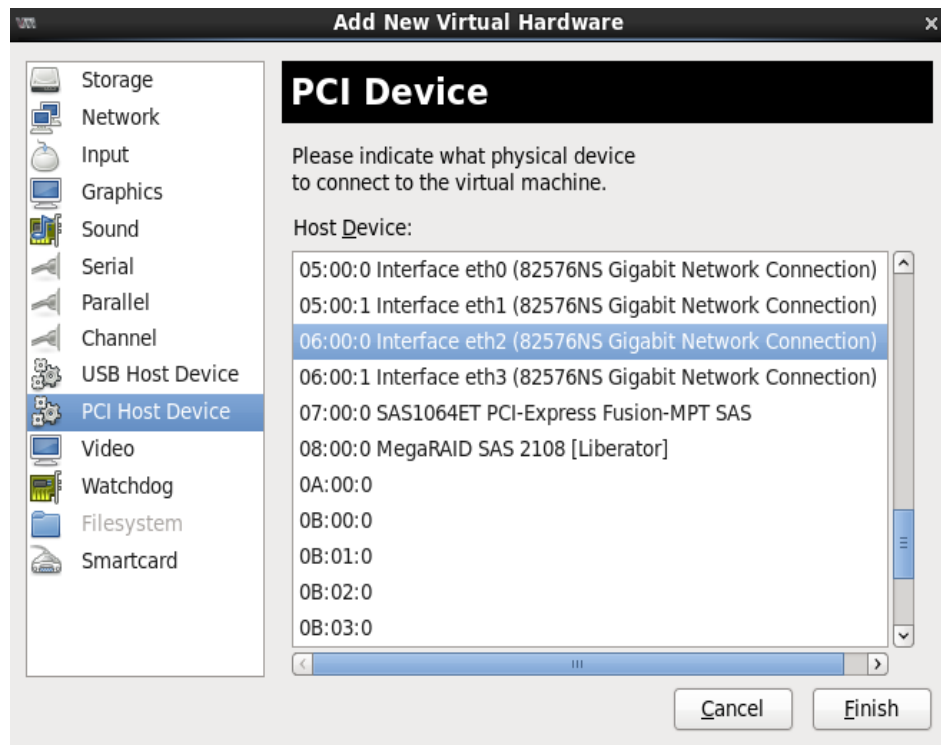
```
# virt-manager
```

2.2.2 为虚拟机添加 PCI 设备

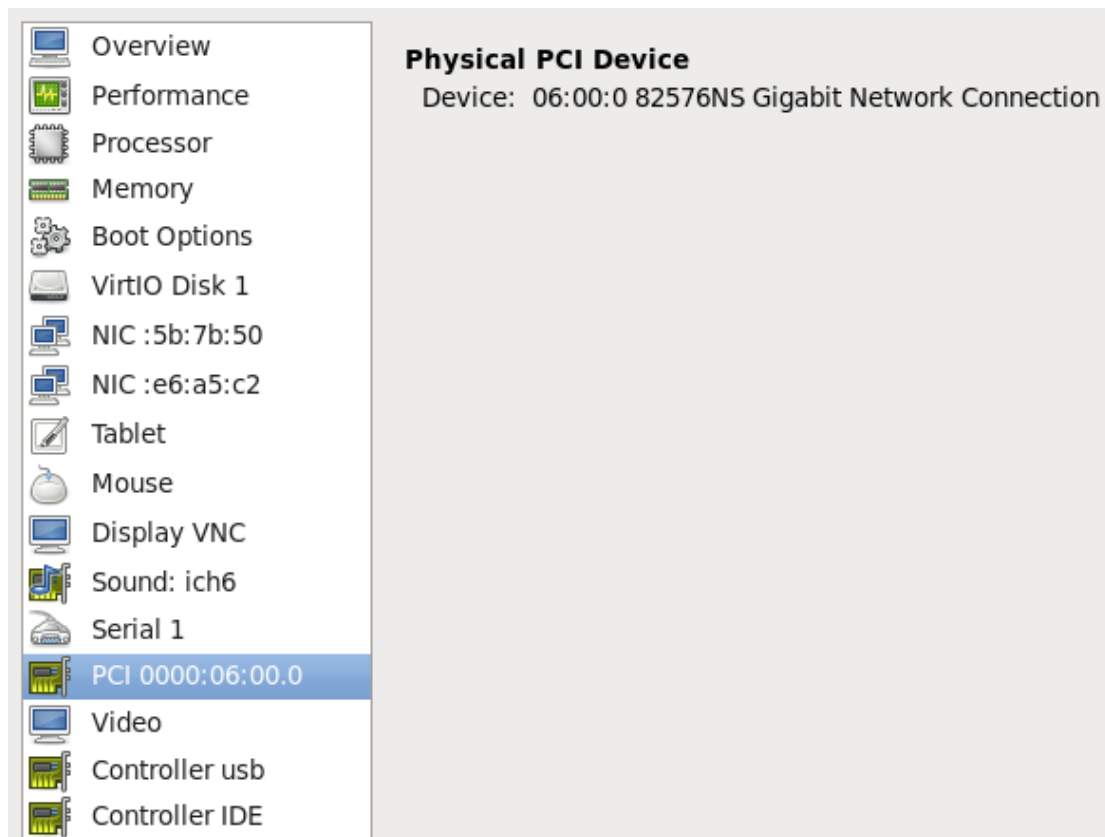
点击 Add Hardware，添加 PCI 设备，如下图所示



2.2.3 选择 PCI 设备



2.2.4 完成 PCI 设备的添加，如下图所示：



2.2.5 启动虚拟机

```
# virsh start 虚拟机名 ※7
```

※7 使用 PCI passthrough 时，不能把 host 全部的资源分配给虚拟机。

2.2.6 用 lspci 确认设备

```
# lspci | grep Ethernet
06:00.0 Ethernet controller: Intel Corporation 82576NS Gigabit Network Connection
(rev 01)
```

2.2.7 确认该设备的驱动器状态

```
# readlink /sys/bus/pci/devices/0000\:00\:06.0/driver
../../../../bus/pci/drivers/igb
```

2.2.8 在虚拟机中，设置 IP 地址

```
# vim /etc/sysconfig/network-scripts/ifcfg-eth1
DEVICE="eth1"
BOOTPROTO="static"
HWADDR="52:54:00:8e:26:de"
NM_CONTROLLED="yes"
ONBOOT="yes"
TYPE="Ethernet"
IPADDR=192.168.2.1 ※9
```

※9 与对联机同一网段的 IP 地址

2.2.8 在虚拟机中，重启网络

```
# /etc/rc.d/init.d/network restart
# ifconfig | grep eth1
```

2.2.9 在虚拟机中，和对联机 ping 确认

```
# ping 对联机的 ip 地址
```

```
# ping 192.168.2.2
PING 192.168.2.2 (192.168.2.2) 56(84) bytes of data.
64 bytes from 192.168.2.2: icmp_seq=1 ttl=64 time=0.337 ms
64 bytes from 192.168.2.2: icmp_seq=2 ttl=64 time=0.285 ms
```