

# qemu-kvm 基线配置手册

内部参考 请勿外传

## 文档属性

客户名称		文件类别	技术文档
文件名	qemu-kvm基线配置手册	是否保密	是
编制者	赵凡	编制者职位	售后工程师
编制者邮箱	zhaofan@kylinos.cn	编制日期	2021-8-4
版本修订记录			
版本号	修订时间	修订说明	
V1.0	2021-8-4	新建	

---

目录

一、 文档说明	4
二、 背景	4
三、 方案	4

内部参考 请勿外传

## 一、文档说明

适用于 kylin v10-sp1 桌面版本 (x86\_64) 配置

## 二、背景

KVM (基于内核的虚拟机) 是内置在 Linux 内核中的开源虚拟化技术。用 KVM 运行多个 Linux 或 Windows 来宾虚拟机。每个来宾都彼此完全隔离, 并具有自己的操作系统和专用的虚拟硬件, 例如 CPU, 内存, 网络接口和存储。

本指南提供有关如何在 kylin v10 sp1 桌面上安装和配置 KVM 的说明。

## 三、方案

要运行带有 2 GB 以上 RAM 的 guest 虚拟机, 必须具有 64 位主机系统。

在继续安装之前, 请确保主机支持 KVM 虚拟化。系统应具有带 VT-x (vmx) 的 intel 处理器或具有 AMD-V (svm) 技术支持的 AMD 处理器。

运行以下 grep 命令以验证您的处理器支持硬件虚拟化:

```
grep -Eoc '(vmx|svm)' /proc/cpuinfo
```

如果 CPU 支持硬件虚拟化, 则该命令将输出一个大于零的数字, 即 CPU 核心的数量。否则, 如果输出是, 0 则表示 CPU 不支持硬件虚拟化。

在某些计算机上, 制造商可能会在 BIOS 中禁用虚拟技术扩展。

要检查 BIOS 中是否启用了 VT, 请使用 kvm-ok 软件包中随附的工具。以超级用户或具有 sudo 特权的用户身份输入以下命令来安装 cpu-checker 包含该 kvm-ok 命令的软件包:

```
sudo apt update
```

```
sudo apt install cpu-checker
```

安装完成后，检查您的系统是否可以运行硬件加速的 KVM 虚拟机：

```
kvm-ok
```

如果未在 BIOS 中禁用处理器虚拟化功能，则输出将如下所示：

```
INFO: /dev/kvm exists
KVM acceleration can be used
```

否则，该命令将打印，并显示一条失败消息以及关于如何启用扩展程序的一条可选消息。启用 AMD-V 或 VT 技术的过程取决于您的主板和处理器类型。请查阅主板文档，以获取有关如何配置系统 BIOS 的信息。

## 在 kylin v10-sp1 上安装 KVM

运行以下命令以安装 KVM 和其他虚拟化管理软件包：

```
sudo apt install qemu-kvm libvirt-daemon-system libvirt-clients bridge-utils virtinst virt-manager
```

- `qemu-kvm` -为 KVM 管理程序提供硬件仿真的软件。
- `libvirt-daemon-system` -用于将 `libvirt` 守护程序作为系统服务运行的配置文件。
- `libvirt-clients` -用于管理虚拟化平台的软件。
- `bridge-utils` -一组用于配置以太网桥的命令行工具。
- `virtinst` -一组用于创建虚拟机的命令行工具。
- `virt-manager` -易于使用的 GUI 界面和支持命令行工具，用于通过 `libvirt` 管理虚拟机。

安装软件包后，`libvirt` 守护程序将自动启动。您可以通过键入以下内容进行验证：

```
sudo systemctl is-active libvirtd
```

输出：

active

为了能够创建和管理虚拟机，您需要将用户添加到“libvirt”和“kvm”组中。为此，请输入：

```
sudo usermod -aG libvirt $USER
sudo usermod -aG kvm $USER
```

\$USER 是一个环境变量，其中包含当前登录用户的名称。

注销并重新登录，以便刷新组成员身份。

## 网络设置

在安装过程中会创建一个名为“virbr0”的网桥。该设备使用 NAT 将来宾计算机连接到外界。

您可以使用该 brctl 工具列出当前网桥及其连接的接口：

```
brctl show
```

输出：

bridge name	bridge id	STP enabled	interfaces
virbr0	8000.52540089db3f	yes	virbr0-nic

“virbr0”网桥未添加任何物理接口。“virbr0-nic”是虚拟设备，没有流量通过该虚拟设备。该设备的唯一目的是避免更改“virbr0”网桥的 MAC 地址。

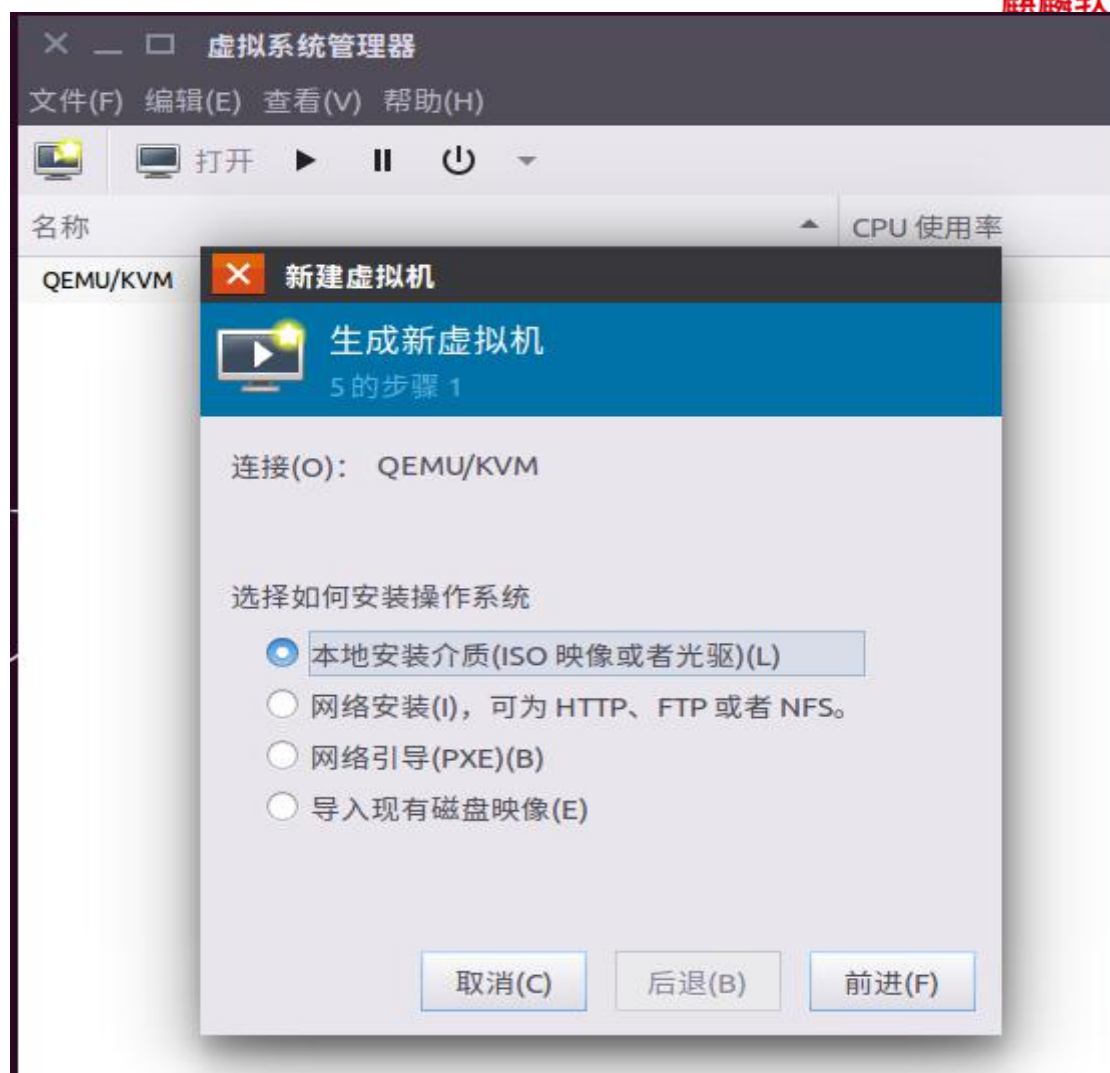
此网络设置适用于大多数桌面用户，但有局限性。如果要从本地网络外部访问来宾，则需要创建一个新的网桥并对其进行配置，以便来宾计算机可以通过主机物理接口连接到外部世界。

## 创建虚拟机

现在，桌面上已安装了 KVM，您可以创建第一个 VM。可以从命令行或使用 virt-manager 应用程序完成此操作。

下载要安装的操作系统的 ISO 映像，然后按照以下步骤创建虚拟机：

- 1, 在所有软件搜索栏中，键入“虚拟机管理器”，然后单击图标以启动该应用程序。
- 2, 启动应用程序后，从顶部菜单中单击“文件”->“新建虚拟机”：



- 3.将会出现一个新窗口。选择“本地安装媒体”，然后单击“转发”按钮。
- 4.提供您的 ISO 映像路径，然后单击“前进”按钮。
- 5.在下一个屏幕中，选择 VM 的内存和 CPU 设置。单击前进。
- 6.接下来，选择“为虚拟机创建磁盘映像”，然后选择 VM 的磁盘空间大小。单击前进。
- 7.输入您的虚拟机名称的名称，然后单击“完成”。
- 8.VM 将启动，并打开一个新窗口：/usr/bin/kvm



从这里，可以按照屏幕上的说明完成操作系统的安装。

## qemu-system-x86\_64 参数介绍

-M	指定仿真机器的种类（通过 -M? 可以显示 qemu 支持的机器种类的列表）
-cpu	指定 CPU 的模型（通过 -cpu? 可以显示可指定的模型的列表）
-smp	指定 CPU 数（未指定时设置为 1）。 smp 选项的子选项如下（用逗号分开指定）： maxcpus=cpus 包括可以动态添加的 CPU 在内的最大 CPU 数 cores=cores 平均 1 个套接字的核数 threads=threads 平均 1 个核的线程数 sockets=sockets 系统的套接字数
-fda/-fdb	分别将文件指定为第一台计算机的软盘、第二台计算机的软盘的映像
-hda/-hdb/-hdc/-hdd	分别将文件指定为第一台计算机、第二台计算机、第三台计算机、第四台计算机的 IDE 硬盘的映像
-cdrom	将文件指定为 IDE CD-ROM 映像（CD-ROM 连接到第二台计算机的 IDE 的主设备）
-drive	详细指定磁盘驱动程序 drive 选项的子选项如下（用逗号分开指定）： file=file 将 file 指定为磁盘映像 if=interface 指定连接驱动器的接口种类（ide、scsi、sd、mtd、floppy、pflash、virtio）



## 绑定共享 L2 缓存的

<b>-drive</b>	<b>bus=bus,unit=unit</b>	指定连接驱动器的总线编号和单元 id
	<b>index=index</b>	指定将驱动器连接到第几个接头
	<b>media=media</b>	指定驱动器的种类 (disk、cdrom)
	<b>cache=cache</b>	指定访问磁盘上的数据时怎样使用主机操作系统上的磁盘缓存 (none、write back、unsafe、writethrough)
	<b>aio=aio</b>	指定 threads 或 native。threads 表示基于 pthread 的非同步 I/O，native 表示 Linux AIO
	<b>format=format</b>	指定磁盘格式。这时不会自动检测磁盘格式，因此，如果指定 format=raw 就可以避免错误运行
	<b>serial=serial</b>	赋予设备的串行号
<b>-boot</b>	<b>addr=addr PCI</b>	指定 PCI 地址 (仅 if=virtio 时)
		指定启动顺序。启动顺序可以指定软盘 (a)、硬盘 (c)、CD-ROM (d)、网络 (n)。
<b>-m</b>		指定内存量 (单位为 MB)
<b>-k</b>		指定键盘布局
<b>-usb</b>		启用 usb
<b>-usbdevice</b>		指定 usb 设备 (mouse、tablet、disk、host、serial、braille、net)
<b>-device</b>		指定添加的设备。可用的驱动程序列表可以通过 -device? 来显示。各驱动程序的可设置属性可以通过 -device driver,? 来显示
<b>-name</b>		设置客户端名称
<b>-uuid</b>		指定机器的 UUID
<b>-nographic</b>		禁用图形模式
<b>-spice</b>		启用 spice 远程桌面协议
<b>-vga</b>		指定显卡的种类 (std、cirrus、vmware、xenfb、qxl、none)
<b>-vnc</b>		开启 VNC 服务器
<b>-enable-kvm</b>		启用 KVM

如创建 512M 内存，2 个 vcpu 的 kvm 虚拟机：

```
qemu-system-x86_64 --enable-kvm -m 512 -smp 2 -name kylin -boot d -hda
/dev/sd*(对应启动分区) -cdrom *.iso -net nic -net tap
```

## 绑定共享 L2 缓存的物理 cpu，提高性能

多核 cpu 在核之间共享 L2 缓存或 L3 缓存。将某个虚拟机的两个虚拟 CPU 分别分配给不共享 L2 缓存的核时，一旦加大内存负载可能造成 cpu 性能降低。

```
cat /sys/devices/system/cpu/cpu0/cache/index1/shared_cpu_list

0,4

#查看 cpu0 L2 缓存共享列表；0,4 号 cpu 共享 L2

virsh vcpupin 虚拟机 name 虚拟 cpu1 物理 cpu0

virsh vcpupin 虚拟机 name 虚拟 cpu2 物理 cpu4

#绑定虚拟 cpu 到共享 L2 缓存的物理 cpu 上
```

## 确认相关硬件虚拟化是否开启

Intel 处理器为例：

MMU 硬件虚拟化 EPT 功能支持，通过 `cat /proc/cpuinfo|grep ept` 查看

IOMMU DMA iO 模式的支持，通过 `dmesg |grep DMAR` 查看

## 通过 KSM 节约内存

KSM (Kernel Samepage Merging)，通过共享相同内容的存储页面，将其整合为一体，从而有效的使用内存功能。

编译内核时，需要设置 `CONFIG_KSM=y`，KSM 通过内核线程 `ksmd` 定期对用户内存进行扫描。

KSM 使用 `/sys/kernel/mm/ksm` 下文件来进行设置：

<code>pages_to_scan</code>	单次扫描的页数。体现在 KSM 通过 <code>MADV_MERGEABLE</code> 指定的页面扫描和确认是否可以合并时的扫描这二者当中
<code>sleep_millisecs</code>	<code>ksmd</code> 在下次扫描前休眠的时间（毫秒）
<code>run</code>	设置为 0~2 的值。0 为停止 <code>ksmd</code> 。到目前为止合并的页仍保持原状。1 为启动 <code>ksmd</code> 。2 为停止 <code>ksmd</code> 。到目前为止合并的页全部生成副本，取消页面共享
<code>pages_shared</code>	共享后的页面数。例如，如果 1000 页均为同一内容，可以合并到 1 页，则 <code>pages_shared</code> 的值为 1
<code>pages_sharing</code>	可共享的页面数。如果 1000 页可以合并为 1 页，则 <code>pages_sharing</code> 的值为 1000
<code>pages_unshared</code>	当前未合并的页数。通过 <code>MADV_MERGEABLE</code> 指定的页且没有同一内容的页，即，不能合并的页数
<code>pages_volatile</code>	输入到 KSM 所管理的树中的页数。KSM 将 <code>MADV_MERGEABLE</code> 指定的页输入专用的管理树中。这个页数是扫描对象的页数，还未确认是否可以合并。经过扫描，确认是否合并后，这个值就会减小
<code>full_scans</code>	从头至尾扫描合并区域的次数

设置 `run` 为 1，尽量调大 `pages_sharing/pages_shared` 比例。

## 对内存设置限制

```
virsh memtune 虚拟机 name
#查看当前虚拟机内存限制设置
virsh memtune 虚拟机 name --hard-limit 2048000 --live
#限制虚拟机最大内存为 200M
```

## IO 优化设置

设置虚拟机 IO 权重，范围 100~1000，权重值越高，IO 优先级越高

```
virsh blkio tune 虚拟机 name
```

```
#查看虚拟机 IO 权重
```

```
virsh blkio tune 虚拟机 name --weight 500
```

```
#设置虚拟机 IO 权重为 500
```