



## Red Hat Virtualization 4.4

# 在 Red Hat Virtualization 中为虚拟机设置 NVIDIA GPU

如何在 Red Hat Virtualization 中配置虚拟机以使用专用的 GPU 或 vGPU。



# Red Hat Virtualization 4.4 在 Red Hat Virtualization 中为虚拟机设置 NVIDIA GPU

---

如何在 Red Hat Virtualization 中配置虚拟机以使用专用的 GPU 或 vGPU。

Red Hat Virtualization Documentation Team

Red Hat Customer Content Services

rhev-docs@redhat.com

## 法律通告

Copyright © 2023 Red Hat, Inc.

The text of and illustrations in this document are licensed by Red Hat under a Creative Commons Attribution–Share Alike 3.0 Unported license ("CC-BY-SA"). An explanation of CC-BY-SA is available at

<http://creativecommons.org/licenses/by-sa/3.0/>

. In accordance with CC-BY-SA, if you distribute this document or an adaptation of it, you must provide the URL for the original version.

Red Hat, as the licensor of this document, waives the right to enforce, and agrees not to assert, Section 4d of CC-BY-SA to the fullest extent permitted by applicable law.

Red Hat, Red Hat Enterprise Linux, the Shadowman logo, the Red Hat logo, JBoss, OpenShift, Fedora, the Infinity logo, and RHCE are trademarks of Red Hat, Inc., registered in the United States and other countries.

Linux<sup>®</sup> is the registered trademark of Linus Torvalds in the United States and other countries.

Java<sup>®</sup> is a registered trademark of Oracle and/or its affiliates.

XFS<sup>®</sup> is a trademark of Silicon Graphics International Corp. or its subsidiaries in the United States and/or other countries.

MySQL<sup>®</sup> is a registered trademark of MySQL AB in the United States, the European Union and other countries.

Node.js<sup>®</sup> is an official trademark of Joyent. Red Hat is not formally related to or endorsed by the official Joyent Node.js open source or commercial project.

The OpenStack<sup>®</sup> Word Mark and OpenStack logo are either registered trademarks/service marks or trademarks/service marks of the OpenStack Foundation, in the United States and other countries and are used with the OpenStack Foundation's permission. We are not affiliated with, endorsed or sponsored by the OpenStack Foundation, or the OpenStack community.

All other trademarks are the property of their respective owners.

## 摘要

本文档论述了如何使用带有图形处理单元(GPU)的主机，在 Red Hat Virtualization 中针对不需要 GPU 的数据密集型任务和软件运行。

---

# 目录

前言 .....	3
<b>第 1 章 GPU 设备透传：将主机 GPU 分配给一个虚拟机</b> .....	<b>4</b>
1.1. 启用主机 IOMMU 支持和将 NOUVEAU 列入黑名单	4
1.2. 从主机中分离 GPU	5
1.3. 将 GPU 附加到虚拟机	6
1.4. 在虚拟机上安装 GPU 驱动程序	6
1.5. 更新和启用 XORG (LINUX 虚拟机)	7
1.6. 从虚拟机中删除主机 GPU	8
<b>第 2 章 分配虚拟 GPU</b> .....	<b>9</b>
2.1. 在主机上设置 NVIDIA vGPU 设备	9
2.2. 在虚拟机上安装 vGPU 驱动程序	10
2.3. 删除 NVIDIA vGPU 设备	11
2.4. 监控 NVIDIA vGPU	11
2.5. 用于 NVIDIA vGPU 的远程桌面流服务	12
<b>第 3 章 相关信息</b> .....	<b>13</b>
<b>附录 A. 法律通知</b> .....	<b>14</b>



## 前言

您可以使用带有兼容图形处理单元(GPU)的主机，在 Red Hat Virtualization 中运行适合图形密集型任务的虚拟机以及运行在没有 GPU（如 CAD）运行的软件。

您可以使用以下方法之一为虚拟机分配 GPU：

- **GPU passthrough**：您可以将主机 GPU 分配给单个虚拟机，因此是虚拟机而不是主机在使用 GPU。
- **虚拟 GPU(vGPU)**：您可以将物理 GPU 设备划分为一个或多个虚拟设备，称为 *介质设备*。然后，您可以将这些介质设备分配给一个或多个虚拟机作为虚拟 GPU。这些虚拟机可以共享单个物理 GPU 的性能。对于某些 GPU，单个虚拟机只能分配一个介质设备。vGPU 支持只在选择的 NVIDIA GPU 上可用。

### 例如：

主机有四个 GPU。每个 GPU 最多支持 16 vGPU，共 64 个 vGPU。以下是一些可能的 vGPU 分配：

- 一台带有 64 个 vGPU 的虚拟机
- 64 台虚拟机，每个虚拟机均具有一个 vGPU
- 32 个虚拟机，每个有一个 vGPU；8 个虚拟机，每个配备 2 个 vGPU；4 个虚拟机，每个具有 4 个 vGPU

## 第 1 章 GPU 设备透传：将主机 GPU 分配给一个虚拟机

Red Hat Virtualization 支持 PCI VFIO（也称为设备透传）作为非VGA 图形设备，适用于某些基于 NVIDIA PCIe 的 GPU 设备。

除了标准模拟图形界面之一外，您可以通过主机 GPU 将一个或多个主机 GPU 附加到单个虚拟机。虚拟机使用仿真图形设备进行预引导和安装，并且 GPU 在加载其图形驱动程序时进行控制。

有关您可以传递给单个虚拟机的具体主机 GPU 数量的信息，请参阅 NVIDIA 网站。

要为虚拟机分配 GPU，请按照以下步骤执行：

1. 在主机机器上启用 I/O 内存管理单元(IOMMU)。
2. 从主机中分离 GPU。
3. 将 GPU 附加到客户端。
4. 在虚拟机上安装 GPU 驱动程序。
5. 在虚拟机上配置 Xorg。

这些步骤的详情如下。

### 先决条件

- 您的 GPU 设备支持 GPU 透传模式。
- 您的系统被列为是已经过验证的有效服务器硬件平台。
- 您的主机芯片组支持 Intel VT-d 或 AMD-Vi

有关支持的硬件和软件的更多信息，请参阅 *NVIDIA GPU 软件发行注册* 中的 [验证的平台](#)。

### 1.1. 启用主机 IOMMU 支持和将 NOUVEAU 列入黑名单

在主机上需要支持 I/O 内存管理单元(IOMMU)在虚拟机上使用 GPU。

#### 流程

1. 在管理门户中，点 **Compute** → **Hosts**。选择主机，再点**编辑**。此时会显示 **Edit Hosts** 窗格。
2. 点 **Kernel** 选项卡。
3. 选中 **Hostdev Passthrough** 和 **SR-IOV** 复选框。此复选框通过在内核命令行中添加 **intel\_iommu=on** 或 **amd\_iommu=on** 来为带有 Intel VT-d 或 AMD Vi 的主机启用 IOMMU 支持。
4. 选中 **Blacklist Nouveau** 复选框。
5. 点击 **确定**。
6. 选择主机并点 **Management** → **Maintenance** 和 **OK**。
7. 点 **Installation** → **Reinstall**。

8. 重新安装完成后，重启主机机器。
9. 主机计算机重新引导后，点 **Management** → **Activate**。



### 注意

要使用命令行启用 IOMMU 支持，请编辑虚拟机中的 **grub.conf** 文件(./entries/rhvh-4.4.<machine id>.conf)使其包含选项 **intel\_iommu=on**。

## 1.2. 从主机中分离 GPU

如果 GPU 绑定到主机内核驱动程序，则无法将 GPU 添加到虚拟机，因此您必须在主机中取消绑定 GPU 设备，然后才能将其添加到虚拟机。主机驱动程序通常不支持 GPU 的动态绑定，因此建议手动将设备从绑定到主机驱动程序中排除。

### 流程

1. 在主机上，运行 **lspci** 命令识别该设备的设备插槽名称和 ID。在以下示例中，使用 NVIDIA Quadro 或 GRID 卡等图形控制器：

```
# lspci -Dnn | grep -i NVIDIA
0000:03:00.0 VGA compatible controller [0300]: NVIDIA Corporation GK104GL [Quadro
K4200] [10de:11b4] (rev a1)
0000:03:00.1 Audio device [0403]: NVIDIA Corporation GK104 HDMI Audio Controller
[10de:0e0a] (rev a1)
```

输出显示安装了 NVIDIA GK104 设备。它具有图形控制器和带有以下属性的音频控制器：

- 图形控制器的设备插槽名称为 **0000:03:00.0**，而 graphics 控制器的 vendor-id:device-id 是 **10de:11b4**。
  - 音频控制器的设备插槽名称为 **0000:03:00.1**，音频控制器的 vendor-id:device-id 是 **10de:0e0a**。
2. 防止主机机器驱动程序使用 GPU 设备。您可以将 vendor-id:device-id 与 pci-stub 驱动程序一起使用。要做到这一点，请将 **pci-stub.ids** 选项（值为 vendor-id:device-id）附加到位于 **/etc/sysconfig/grub** 配置文件中的 **GRUB\_CMDLINX\_LINUX** 环境变量，例如：

```
GRUB_CMDLINE_LINUX="crashkernel=auto resume=/dev/mapper/vg0-lv_swap
rd.lvm.lv=vg0/lv_root rd.lvm.lv=vg0/lv_swap rhgb quiet intel_iommu=on pci-
stub.ids=10de:11b4,10de:0e0a"
```

为 pci-stub 添加附加厂商 ID 和设备 ID 时，使用逗号分隔它们。

3. 使用 **grub2-mkconfig** 重新生成引导装载程序配置，使其包含这个选项：

```
# grub2-mkconfig -o /etc/grub2.cfg
```



### 注意

在使用基于 UEFI 的主机时，目标文件应为 **/etc/grub2-efi.cfg**。

4. 重启主机机器。

5. 确认启用了 IOMMU，主机设备被添加到 pci-stub.ids 列表中，并且 Nouveau 已放入黑名单：

```
# cat /proc/cmdline
BOOT_IMAGE=(hd0,msdos1)/vmlinuz-4.18.0-147.el8.x86_64 root=/dev/mapper/vg0-lv_root
ro crashkernel=auto resume=/dev/mapper/vg0-lv_swap rd.lvm.lv=vg0/lv_root
rd.lvm.lv=vg0/lv_swap rhgb quiet intel_iommu=on 1
pci-stub.ids=10de:11b4,10de:0e0a 2
rdblacklist=nouveau 3
```

- 1** IOMMU 被启用
- 2** 主机设备被添加到 pci-stub.ids 列表中
- 3** Nouveau 将列入黑名单

### 1.3. 将 GPU 附加到虚拟机

从主机内核驱动程序取消绑定 GPU 后，您可以将其添加到虚拟机并启用正确的驱动程序。

#### 流程

1. 按照 [虚拟机管理指南](#) 中的 [将主机设备添加到虚拟机](#) 中的步骤进行操作。
2. 运行虚拟机并登录该虚拟机。
3. 在虚拟机上安装 NVIDIA GPU 驱动程序。
4. 使用 `lspci -nnk` 命令，验证是否为 GPU 使用了正确的内核驱动程序。例如：

```
# lspci -nnk
00:07.0 VGA compatible controller [0300]: NVIDIA Corporation GK104GL [Quadro K4200]
[10de:11b4] (rev a1)
Subsystem: Hewlett-Packard Company Device [103c:1096]
Kernel driver in use: nvidia
Kernel modules: nouveau, nvidia_drm, nvidia
```

### 1.4. 在虚拟机上安装 GPU 驱动程序

#### 流程

1. 运行虚拟机并使用 VNC 或 SPICE 控制台与其连接。
2. 将驱动程序下载到虚拟机。有关获取驱动程序的详情，请查看 [NVIDIA 网站上的驱动程序页面](#)。
3. 安装 GPU 驱动程序。



#### 重要

**仅限 Linux：** 在 Linux 客户机操作系统上安装驱动程序时，系统将提示您更新 `xorg.conf`。如果您不在安装过程中更新 `xorg.conf`，则需要手动更新它。

4. 在驱动程序完成安装后，重启机器。对于 **Windows 虚拟机**，从管理门户或虚拟机门户中完全关闭客户机，而不是从客户机操作系统中。



### 重要

仅 **Windows**：从 Windows 虚拟机操作系统中关闭虚拟机有时会将虚拟机变为休眠模式，这不会完全清除内存，并可能会导致后续的问题。使用管理门户或虚拟机门户关闭虚拟机会强制完全清理内存。

5. 将监视器连接到主机 GPU 输出接口，并运行虚拟机。
6. 为每个 vGPU 设置 NVIDIA vGPU 客户机软件许可，并在 NVIDIA 控制面板中添加许可证凭证。如需更多信息，请参阅 *NVIDIA Virtual GPU 软件文档* 中的 [NVIDIA vGPU 软件许可证的使用方法](#)。

## 1.5. 更新和启用 XORG（LINUX 虚拟机）

在在虚拟机上使用 GPU 之前，您需要在虚拟机上更新并启用 xorg。NVIDIA 驱动程序安装应该自动执行此操作。查看 `/etc/X11/xorg.conf` 来检查 xorg 是否已更新并启用：

```
# cat /etc/X11/xorg.conf
```

前两行表示是否由 NVIDIA 生成。例如：

```
# cat /etc/X11/xorg.conf
# nvidia-xconfig: X configuration file generated by nvidia-xconfig
# nvidia-xconfig: version 390.87 (buildmeister@swio-display-x64-rhel04-14) Tue Aug 21 17:33:38
PDT 2018
```

### 流程

1. 在虚拟机上，使用以下命令生成 `xorg.conf` 文件：

```
# X -configure
```

2. 使用以下命令，将 `xorg.conf` 文件复制到 `/etc/X11/xorg.conf`：

```
# cp /root/xorg.conf.new /etc/X11/xorg.conf
```

3. 重启虚拟机。
4. 通过查看 `/etc/X11/xorg.conf` 来验证 xorg 是否已更新并启用：

```
# cat /etc/X11/xorg.conf
```

搜索 **Device** 部分。您应该看到类似如下的条目：

```
Section "Device"
    Identifier   "Device0"
    Driver       "nvidia"
    VendorName   "NVIDIA Corporation"
EndSection
```

GPU 现在分配到虚拟机。

## 1.6. 从虚拟机中删除主机 GPU

- 有关从虚拟机中删除主机 GPU 的详情，请参考 [虚拟机管理指南](#) 中的 [从虚拟机中删除主机设备](#)。

## 第 2 章 分配虚拟 GPU

要设置 NVIDIA vGPU 设备，您需要：

1. 为您的 GPU 设备获取并安装正确的 NVIDIA vGPU 驱动程序
2. 创建介质设备
3. 为虚拟机分配每个介质设备
4. 在每个虚拟机上安装客户机驱动程序。

以下流程解释了此过程。

### 2.1. 在主机上设置 NVIDIA vGPU 设备



#### 注意

在客户端操作系统中安装 NVIDIA vGPU 驱动程序前，您需要理解许可要求并获取正确的许可证凭证。

#### 先决条件

- 您的 GPU 设备支持虚拟 GPU(vGPU)功能。
- 您的系统被列为是已经过验证的有效服务器硬件平台。

有关支持的 GPU 和验证平台的更多信息，请参阅 [www.nvidia.com](http://www.nvidia.com) 上的 [NVIDIA vGPU CERTIFIED SERVERS](#)。

#### 流程

1. 下载并安装 NVIDIA-vGPU 驱动程序。有关获取驱动程序的详情，请参考 [NVIDIA 网站中的 vGPU 驱动程序页面](#)。要下载驱动程序，需要一个 Nvidia enterprise 帐户。如果硬件厂商不可用，请联系该厂商。
2. 从 Nvidia 网站解压缩下载的文件并将其复制到主机以安装驱动程序。
3. 如果 NVIDIA 软件安装程序没有创建 `/etc/modprobe.d/nvidia-installer-disable-nouveau.conf` 文件，请手动创建。
4. 在文本编辑器中打开 `/etc/modprobe.d/nvidia-installer-disable-nouveau.conf` 文件，并在文件末尾添加以下行：

```
blacklist nouveau
options nouveau modeset=0
```

5. 为当前内核重新生成初始 ramdisk，然后重新启动：

```
# dracut --force
# reboot
```

另外，如果您需要使用带有介质设备的之前支持的内核版本，请为所有安装的内核版本重新生成初始 ramdisk：

```
# dracut --regenerate-all --force
# reboot
```

- 检查内核是否已载入 `nvidia_vgpu_vfio` 模块：

```
# lsmod | grep nvidia_vgpu_vfio
```

- 检查 `nvidia-vgpu-mgr.service` 服务是否正在运行：

```
# systemctl status nvidia-vgpu-mgr.service
```

例如：

```
# lsmod | grep nvidia_vgpu_vfio
nvidia_vgpu_vfio 45011 0
nvidia 14333621 10 nvidia_vgpu_vfio
mdev 20414 2 vfio_mdev,nvidia_vgpu_vfio
vfio 32695 3 vfio_mdev,nvidia_vgpu_vfio,vfio_iommu_type1
# systemctl status nvidia-vgpu-mgr.service
nvidia-vgpu-mgr.service - NVIDIA vGPU Manager Daemon
   Loaded: loaded (/usr/lib/systemd/system/nvidia-vgpu-mgr.service; enabled; vendor preset: disabled)
   Active: active (running) since Fri 2018-03-16 10:17:36 CET; 5h 8min ago
   Main PID: 1553 (nvidia-vgpu-mgr)
   [...]

```

- 在管理门户中，点 **Compute** → **Virtual Machines**。
- 单击虚拟机的名称，以转至详情视图。
- 单击 **主机设备** 选项卡。
- 点 **Manage vGPU**。此时会打开 **Manage vGPU** 对话框。
- 选择一个 vGPU 类型以及要与这个虚拟机搭配使用的实例数量。
- 选择 **On** 用于 **Secondary display adapter for VNC**，在 vGPU 之外，为控制台添加第二个模拟 QXL 或 VGA 图形适配器作为控制台的主要图形适配器。



### 注意

在集群级别 4.5 及更新的版本中，当使用 vGPU 且 **Secondary display adapter for VNC** 被设置为 **On**，则会自动向虚拟机添加一个额外的帧缓冲显示设备。这允许在进行 vGPU 初始化前显示虚拟机控制台，而不是空白屏幕。

- 单击 **Save**。

## 2.2. 在虚拟机上安装 vGPU 驱动程序

### 流程

- 运行虚拟机并使用 VNC 控制台连接到该虚拟机。



### 注意

vGPU 不支持 SPICE。

2. 将驱动程序下载到虚拟机。有关获取驱动程序的详情，请查看 [NVIDIA 网站上的驱动程序页面](#)。
3. 安装 vGPU 驱动程序，按照 *NVIDIA Virtual GPU 软件文档* 中的 [安装 NVIDIA vGPU Software Graphics Driver](#) 中的说明。



### 重要

**仅限 Linux：**在 Linux 客户机操作系统上安装驱动程序时，系统将提示您更新 xorg.conf。如果您不在安装过程中更新 xorg.conf，则需要手动更新它。

4. 在驱动程序完成安装后，重启机器。对于 **Windows 虚拟机**，从管理门户或虚拟机门户中完全关闭客户机，而不是从客户机操作系统中。



### 重要

**仅 Windows：**从 Windows 虚拟机操作系统中关闭虚拟机有时会将虚拟机变为休眠模式，这不会完全清除内存，并可能会导致后续的问题。使用管理门户或虚拟机门户关闭虚拟机会强制完全清理内存。

5. 运行虚拟机并使用其中一个支持的远程桌面协议（如 Mechdyne TGX）连接到该虚拟机，并通过打开 NVIDIA Control Panel 进行验证。在 Windows 上，您还可以打开 Windows Device Manager。vGPU 应显示在 **Display 适配器** 下。如需更多信息，请参阅 *NVIDIA Virtual GPU 软件文档* 中的 [NVIDIA vGPU 软件图形驱动程序](#)。
6. 为每个 vGPU 设置 NVIDIA vGPU 客户机软件许可，并在 NVIDIA 控制面板中添加许可证凭证。如需更多信息，请参阅 *NVIDIA Virtual GPU 软件文档* 中的 [NVIDIA vGPU 软件许可证的使用方法](#)。

## 2.3. 删除 NVIDIA vGPU 设备

要更改分配的 vGPU 介质设备的配置，必须将现有设备从分配的客户端中删除。

### 流程

1. 在管理门户中，点 **Compute → Virtual Machines**。
2. 单击虚拟机的名称，以转至详情视图。
3. 单击 **主机设备** 选项卡。
4. 点 **Manage vGPU**。此时会打开 **Manage vGPU** 对话框。
5. 单击 **Selected vGPU Type Instances** 旁边的 **x** 按钮，从虚拟机中分离 vGPU。
6. 单击 **SAVE**。

## 2.4. 监控 NVIDIA vGPU

对于 NVIDIA vGPU，若要获取物理 GPU 和 vGPU 的信息，您可以通过在主机上输入 **nvidia-smi** 命令来使用 NVIDIA System Management Interface。如需更多信息，请参阅 *NVIDIA Virtual GPU 软件文档* 中的 [NVIDIA 系统管理接口 nvidia-smi](#)。

例如：

```
# nvidia-smi
Thu Nov 1 17:40:09 2018
+-----+
| NVIDIA-SMI 410.62                Driver Version: 410.62          |
+-----+-----+
| GPU Name      Persistence-M| Bus-Id        Disp.A | Volatile Uncorr. ECC |
| Fan  Temp  Perf  Pwr:Usage/Cap|  Memory-Usage | GPU-Util  Compute M. |
+-----+-----+-----+-----+-----+-----+
| 0  Tesla M60      On      | 00000000:84:00.0 Off |                    |
| N/A   40C    P8   24W / 150W | 1034MiB / 8191MiB | 0%      Default |
+-----+-----+-----+-----+-----+
| 1  Tesla M60      On      | 00000000:85:00.0 Off |                    |
| N/A   33C    P8   23W / 150W | 8146MiB / 8191MiB | 0%      Default |
+-----+-----+-----+-----+-----+
| 2  Tesla M60      On      | 00000000:8B:00.0 Off |                    |
| N/A   34C    P8   24W / 150W | 8146MiB / 8191MiB | 0%      Default |
+-----+-----+-----+-----+-----+
| 3  Tesla M60      On      | 00000000:8C:00.0 Off |                    |
| N/A   45C    P8   24W / 150W | 18MiB / 8191MiB | 0%      Default |
+-----+-----+-----+-----+-----+

+-----+
| Processes:                         GPU Memory |
| GPU   PID  Type  Process name      Usage   |
+-----+-----+-----+-----+-----+
| 0    34432  C+G  vgpu              508MiB |
| 0    34718  C+G  vgpu              508MiB |
| 1    35032  C+G  vgpu              8128MiB |
| 2    35032  C+G  vgpu              8128MiB |
+-----+-----+-----+-----+-----+
```

## 2.5. 用于 NVIDIA vGPU 的远程桌面流服务

以下远程桌面流服务已被成功测试，以便在 RHEL 8 中与 NVIDIA vGPU 功能一起使用：

- HP-RGS
- Mechdyne TGX - 目前无法在 Windows Server 2016 客户机中使用 Mechdyne TGX。
- NICE DCV - 使用此流服务时，使用固定分辨率设置，因为在某些情况下使用动态解析会导致黑屏。

---

## 第 3 章 相关信息

有关在带有 KVM 的 RHEL 上使用 NVIDIA vGPU 的详情，请参考：

- [NVIDIA GPU 软件发行说明](#)。
- *NVIDIA 虚拟 GPU 软件文档* <https://docs.nvidia.com>。

## 附录 A. 法律通知

Copyright © 2022 Red Hat, Inc.

Licensed under the ([Creative Commons Attribution–ShareAlike 4.0 International License](#)).从(oVirt Project)的文档衍生而来。如果您发布本文档或对其进行改编，您必须提供原始版本的 URL。

修改后的版本必须删除所有红帽商标。

Red Hat、Red Hat Enterprise Linux、Red Hat 商标、Shadowman 商标、JBoss、OpenShift、Fedora、Infinity 商标以及 RHCE 都是在美国及其他国家的注册商标。

Linux® 是 Linus Torvalds 在美国和其他国家/地区的注册商标。

Java® 是 Oracle 和/或其附属公司的注册商标。

XFS® 是 Silicon Graphics International Corp. 或其子公司在美国和/或其他国家的商标。

MySQL® 是 MySQL AB 在美国、欧盟和其他国家/地区的注册商标。

Node.js® 是 Joyent 的官方商标。Red Hat Software Collections 与官方 Joyent Node.js 开源或商业项目没有正式关联或被正式认可。

The OpenStack® Word Mark 和 OpenStack 标识是 OpenStack Foundation 在美国及其他国家的注册商标/服务标记或商标/服务标记，可根据 OpenStack Foundation 授权使用。我们不附属于 OpenStack Foundation 或 OpenStack 社区。

所有其他商标均由其各自所有者所有。