



美国硅谷第二可用区开放

10余款云产品上线

全美双可用区尽享ECS88折, 10.10~11.9 仅此一月

[查看详情](#)[🏠](#) » [技术](#) » [学习](#) » [查看内容](#)

RHCSA 系列（十五）：虚拟化基础和使用 KVM 进行虚拟机管理

2015-10-7 10:30 评论: 4 收藏: 5

原文: <http://www.tecmint.com/kvm-virtualization-basics-and-guest-administration/>
译文: LCTT <https://linux.cn/article-6362-1.html>作者: Gabriel Cánepa
译者: FSSlc

假如你在词典中查一下单词“^{virtualize}虚拟化”，你将会发现它的意思是“创造某些事物的一个^{virtualization}虚拟物（而非真实的）”。在计算机行业中，术语^{virtualization}虚拟化指的是：在相同的物理（硬件）系统上，同时运行多个操作系统，且这几个系统相互隔离的^{host}可能性，而那个硬件在虚拟化架构中被称作^{host}宿主机。

RHCSA 系列：虚拟化基础和使用 KVM 进行虚拟机管理 - Part 15

通过使用^{hypervisor}虚拟机监视器（也被称为^{hypervisor}虚拟机管理程序），虚拟机（被称为 guest）由底层的硬件来供给虚拟资源（举几个例子来说，如 CPU，RAM，存储介质，网络接口等）。

考虑到这一点就可以清楚地看出，虚拟化的主要优点是节约成本（在设备和网络基础设施，及维护工作等方面）和显著地减少容纳所有必要硬件所需的物理空间。

由于这个简单的指南不能涵盖所有的虚拟化方法，我鼓励你参考在总结部分中列出的文档，以此对这个话题做更深入的了解。

请记住当前文章只是用于在 RHEL 7 中用命令行工具使用 **KVM** <http://www.linux-kvm.org/page/Main_Page> (^{基于内核的虚拟机}Kernel-based Virtual Machine) 学习虚拟化基础知识的一个起点，而并不是对这个话题的深入探讨。

检查硬件要求并安装软件包

为了设置虚拟化，你的 CPU 必须能够支持它。你可以使用下面的命令来查看你的系统是否满足这个要求：

```
1. | # grep -E 'svm|vmx' /proc/cpuinfo
```

在下面的截图中，我们可以看到当前的系统（带有一个 AMD 的微处理器）支持虚拟化，svm 字样的存在暗示了这一点。假如我们有一个 Intel 系列的处理器，我们将会看到上面命令的结果将会出现 vmx 字样。

检查 KVM 支持

另外，你需要在你宿主机的硬件（BIOS 或 UEFI）中开启虚拟化。

现在，安装必要的软件包：

- qemu-kvm 是一个开源的虚拟机程序，为 KVM 虚拟机监视器提供硬件仿真，而 qemu-img 则提供了一个操纵磁盘镜像的命令行工具。
- libvirt 包含与操作系统的虚拟化功能交互的工具。
- libvirt-python 包含一个模块，它允许用 Python 写的应用来使用由 libvirt 提供的接口。
- libguestfs-tools 包含各式各样的针对虚拟机的系统管理员命令行工具。

- virt-install 包含针对虚拟机管理的其他命令行工具。

命令如下：

```
1. # yum update && yum install qemu-kvm qemu-img libvirt libvirt-python libguestfs-tools virt-install
```

一旦安装完成，请确保你启动并开启了 libvirtd 服务：

```
1. # systemctl start libvirtd.service
2. # systemctl enable libvirtd.service
```

默认情况下，每个虚拟机将只能与放在相同的物理服务器上的虚拟机以及宿主机自身通信。要使得虚拟机能够访问位于局域网或因特网中的其他机器，我们需要像下面这样在我们的宿主机上设置一个桥接接口（比如说 br0）：

1、添加下面的一行到我们的 NIC 主配置中（类似 `/etc/sysconfig/network-scripts/ifcfg-enp0s3` 这样的文件）：

```
1. BRIDGE=br0
```

2、使用下面的内容（注意，你可能需要更改 IP 地址，网关地址和 DNS 信息）为 br0 创建一个配置文件（`/etc/sysconfig/network-scripts/ifcfg-br0`）：

```
1. DEVICE=br0
2. TYPE=Bridge
3. BOOTPROTO=static
4. IPADDR=192.168.0.18
5. NETMASK=255.255.255.0
6. GATEWAY=192.168.0.1
7. NM_CONTROLLED=no
8. DEFROUTE=yes
9. PEERDNS=yes
10. PEERROUTES=yes
11. IPV4_FAILURE_FATAL=no
12. IPV6INIT=yes
13. IPV6_AUTOCONF=yes
14. IPV6_DEFROUTE=yes
15. IPV6_PEERDNS=yes
16. IPV6_PEERROUTES=yes
17. IPV6_FAILURE_FATAL=no
18. NAME=br0
19. ONBOOT=yes
20. DNS1=8.8.8.8
21. DNS2=8.8.4.4
```

3、最后在文件 `/etc/sysctl.conf` 中设置：

```
1. net.ipv4.ip_forward = 1
```

来开启包转发并加载更改到当前的内核配置中：

```
1. # sysctl -p
```

注意，你可能还需要告诉 `firewalld` 让这类的流量应当被允许通过防火墙。假如你需要这样做，记住你可以参考这个系列的 [使用 `firewalld` 和 `iptables` 来控制网络流量](https://linux.cn/article-6315-1.html) <<https://linux.cn/article-6315-1.html>>。

创建虚拟机镜像

默认情况下，虚拟机镜像将会被创建到 `/var/lib/libvirt/images` 中，且强烈建议你不要更改这个设定，除非你真的需要那么做且知道你在做什么，并能自己处理有关 SELinux 的设定（这个话题已经超出了本教程的讨论范畴，但你可以参考这个系列的第 13 部分 [使用 SELinux 来进行强制访问控制](https://linux.cn/article-6339-1.html) <<https://linux.cn/article-6339-1.html>>，假如你想更新你的知识的话）。

这意味着你需要确保你在文件系统中分配了必要的空间来容纳你的虚拟机。

下面的命令将使用位于 `/home/gacanepa/ISOs` 目录下的 `rhel-server-7.0-x86_64-dvd.iso` 镜像文件和 `br0` 这个网桥来创建一个名为 `tecmin-virt01` 的虚拟机，它有一个虚拟 CPU，1 GB（=1024 MB）的 RAM，20 GB 的磁盘空间（由 `/var/lib/libvirt/images/tecmin-virt01.img` 所代表）：

```
1. # virt-install \
2. --network bridge=br0
3. --name tecmin-virt01 \
4. --ram=1024 \
5. --vcpus=1 \
6. --disk path=/var/lib/libvirt/images/tecmin-virt01.img,size=20 \
7. --graphics none \
8. --cdrom /home/gacanepa/ISOs/rhel-server-7.0-x86_64-dvd.iso
9. --extra-args="console=tty0 console=ttyS0,115200"
```

假如安装文件位于一个 HTTP 服务器上，而不是存储在你磁盘中的镜像中，你必须将上面的 `-cdrom` 替换为 `-location`，并明确地指出在线存储仓库的地址。

至于上面的 `-graphics none` 选项，它告诉安装程序只以文本模式执行安装过程。假如你使用一个 GUI 界面和一个 VNC 窗口来访问主虚拟机控制台，则可以省略这个选项。最后，使用 `-extra-args` 参数，我们将传递内核启动参数给安装程序，以此来设置一个串行的虚拟机控制台。

现在，所安装的虚拟机应当可以作为一个正常的（真实的）服务来运行了。假如没有，请查看上面列出的步骤。

管理虚拟机

作为一个系统管理员，还有一些典型的管理任务需要你在虚拟机上去完成。注：下面所有的命令都需要在你的宿主机上运行：

1. 列出所有的虚拟机：

```
1. # virsh list --all
```

你必须留意上面命令输出中的虚拟机 ID（尽管上面的命令还会返回虚拟机的名称和当前的状态），因为你需要它来执行有关某个虚拟机的大多数管理任务。

2. 显示某个虚拟机的信息：

```
1. # virsh dominfo [VM Id]
```

3. 开启，重启或停止一个虚拟机操作系统：

```
1. # virsh start | reboot | shutdown [VM Id]
```

4. 假如网络无法连接且在宿主机上没有运行 X 服务器，可以使用下面的命令来访问虚拟机的串行控制台：

```
1. # virsh console [VM Id]
```

注：这需要你添加一个串行控制台配置信息到 `/etc/grub.conf` 文件中（参考刚才创建虚拟机时传递给 `-extra-args` 选项的参数）。

5. 修改分配的内存或虚拟 CPU：

首先，关闭虚拟机：

```
1. # virsh shutdown [VM Id]
```

为 RAM 编辑虚拟机的配置：

```
1. # virsh edit [VM Id]
```

然后更改

```
1. <memory>[内存大小，注意不要加上方括号]</memory>
```

使用新的设定重启虚拟机：

```
1. # virsh create /etc/libvirt/qemu/tecmin-virt01.xml
```

最后，可以使用下面的命令来动态地改变内存的大小：

```
1. # virsh setmem [VM Id] [内存大小，这里没有括号]
```

对于 CPU，使用：

```
1. # virsh edit [VM Id]
```

然后更改

```
1. <cpu>[CPU 数目，这里没有括号]</cpu>
```

至于更深入的命令和细节，请参考 RHEL 5 虚拟化指南（这个指南尽管有些陈旧，但包括了用于管理虚拟机的 `virsh` 命令的详尽清单）的第 26 章里的表 26.1。

总结

在这篇文章中，我们涵盖了在 RHEL 7 中如何使用 KVM 和虚拟化的一些基本概念，这个话题是一个广泛且令人着迷的话题。并且我希望它能成为你在随后阅读官方的 [RHEL 虚拟化入门 <https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/7/html/Virtualization_Getting_Started_Guide/index.html>](https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/7/html/Virtualization_Getting_Started_Guide/index.html) 和 [RHEL 虚拟化部署和管理指南 <https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/7/html/Virtualization_Deployment_and_Administration_Guide/index.html>](https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/7/html/Virtualization_Deployment_and_Administration_Guide/index.html)，探索更高级的主题时的起点教程，并给你带来帮助。

另外，为了分辨或拓展这里解释的某些概念，你还可以参考先前包含在 **KVM 系列** <<http://www.tecmint.com/install-and-configure-kvm-in-linux/>> 中的文章。

via: <http://www.tecmint.com/kvm-virtualization-basics-and-guest-administration/> <<http://www.tecmint.com/kvm-virtualization-basics-and-guest-administration/>>

作者：[Gabriel Cánepa](http://www.tecmint.com/author/gacanepa/) <<http://www.tecmint.com/author/gacanepa/>> 译者：[FSSlc](https://github.com/FSSlc) <<https://github.com/FSSlc>> 校对：[wxy](https://github.com/wxy) <<https://github.com/wxy>>

本文由 [LCTT](https://github.com/LCTT/TranslateProject) <<https://github.com/LCTT/TranslateProject>> 原创翻译，**Linux中国** <<file:///root/github/my-notes/Manual/Linux.cn/RHCSA/RHCSA%E7%B3%BB%E5%88%9715%E8%99%9A%E6%8B%9F%E5%8C%96%E5%9F%BA%E7%A1%80%E5%92%8C%E4%BD%BF%E7%94%A8%20KVM%20%E8%BF%9B.html>> 荣誉推出

原文：<http://www.tecmint.com/kvm-virtualization-basics-and-guest-administration/> <<http://www.tecmint.com/kvm-virtualization-basics-and-guest-administration/>>

译文：[LCTT](http://lctt.github.io/) <<http://lctt.github.io/>> <https://linux.cn/article-6362-1.html> <<https://linux.cn/article-6362-1.html>>

作者：Gabriel Cánepa

译者：FSSlc

发表评论

验证码 换一个



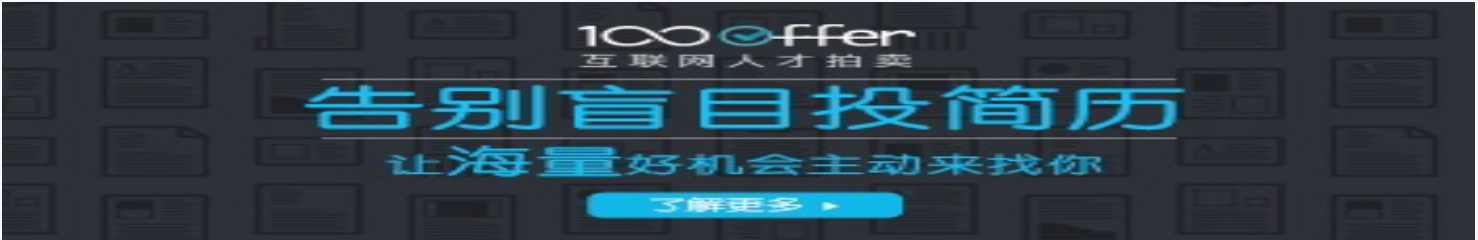
热点评论

linux [Chrome 45.0|Mac 10.11] 2015-10-16 20:05

啊哈，还没发呢。。我这就准备发了，一个系列。

赞 1

体验环境



本文导航

- 检查硬件要求并安装软件包
- 创建虚拟机镜像
- 管理虚拟机
- 总结

相关阅读

• RHCSA 系列（十四）：在 RHEL 7 中设置基于 LDAP 的认证	2015-10-4
• RHCSA 系列（六）：使用 Parted 和 SSM 来配置和加密系统存储	2015-9-21
• RHCSA 系列（七）：使用 ACL（访问控制列表）和挂载 Samba/NFS 共享	2015-9-22
• RHCSA 系列（八）：加固 SSH，设定主机名及启用网络服务	2015-9-23
• RHCSA 系列（九）：安装、配置及加固一个 Web 和 FTP 服务器	2015-9-24
• RHCSA 系列（十）：Yum 包管理、Cron 自动任务计划和监控系统日志	2015-9-26