



开源虚拟化：KVM 介绍

开源虚拟化：KVM 介绍

Linux 虚拟化技术的用户目前有两种免费的开源管理程序可以选择：Xen 和 KVM。KVM 是一个轻量级的虚拟化管理程序模块，该模块主要来自于 Linux 内核。由于其性能和实施的简易性，以及对 Linux 重量级的持续支持，所以还是有必要认真对待 KVM 技术的。

了解 KVM

最近有一些 Linux 厂商——包括 Red Hat 和 Canonical 把基于内核的虚拟机（KVM：Kernel-based Virtual Machine）技术内置在 Ubuntu 中。KVM 到底具有何种吸引力？

- ❖ 虚拟面面观：您该选择 Xen 还是 KVM？
- ❖ KVM 再次引爆 Type 1 与 Type 2hypervisor 战争
- ❖ Red Hat vs. Ubuntu KVM 虚拟化：谁最好？

红帽与 KVM

随着最新版操作系统的发布，红帽公司也放弃了以开源 Xen 为虚拟化平台的思路。Red Hat Enterprise Linux 6 开始支持 KVM（Kernel-based Virtual Machine）作为 hypervisor。可能有人会惧怕迁移过程，实际上 Xen-to-KVM 迁移并不难。

- ❖ 从 Xen 到 KVM：弄清 RHEL 6 的变化
- ❖ RHEL6 里 KVM 网络运行的两种方式
- ❖ KVM 虚拟化：你用 RHEV 还是 RHEL？

KVM 技巧

本部分介绍 KVM 的一些实用技巧。

- ❖ KVM 三大存储选项使用说明
- ❖ 如何调整 KSM 提升 KVM 性能?
- ❖ 通过 virsh 执行 KVM 流程化管理

虚拟面面观：您该选择 Xen 还是 KVM？

Linux 虚拟化技术的用户目前有两种免费的开源管理程序可以选择：Xen 和 KVM。

作为较早出现的虚拟化技术，Xen 是“第一类”运行在裸机上的虚拟化管理程序（Hypervisor），也是当前相当一部分商业化运作公司的基础技术，其中包括 Citrix 系统公司的 XenServer 和 Oracle 的虚拟机。Xen 技术的倡导者们声称 Xen 的性能强劲，并且拥有一个广泛的管理工具和能够交付卓越性能，以及其它诸多优点。

但是最近有一些 Linux 厂商——包括 Red Hat 和 Canonical 把基于内核的虚拟机（KVM: Kernel-based Virtual Machine）技术内置在 Ubuntu 中。KVM 是一个轻量级的虚拟化管理程序模块，该模块主要来自于 Linux 内核。虽然只是后来者，但是由于其性能和实施的简易性，以及对 Linux 重量级的持续支持，所以还是有必要认真对待 KVM 技术的。

在本期虚拟化面面观中，两位 IT 专家将会对 Xen 和 KVM 做出比较，并给出各自的理由。

Andi Mann: Xen 比 KVM 优越的六大原因

Sander van Vugt: KVM 比 Xen 能够带来更好的 Linux 集成

Xen 比 KVM 优越的六大原因

投稿人: Andi Mann

作为开源的虚拟化技术，对比 Xen 和 KVM 可以看到，Xen 以 6 个无与伦比的优势领先：更好的可用资源、平台支持、可管理性、实施、动态迁移和性能基准。

- **可用资源：**Xen 的问世要比 KVM 早 4 年之久（两者分别是 2003 年和 2007 年）。随着 Citrix、Novell、Oracle、Sun、Red Hat 和 Virtual Iron 等公司在市场领域的实施，就比较容易找到精通 Xen 的 IT 技术人员，更容易让这些技术人员接受 Xen 相关的培训、更容易得到 Xen 的咨询帮助以及获得 Xen 证书。企业管理协会（EMA: Enterprise Management Associates）2008 年这对虚拟化和管理的研究报告表明，这些关键因素占到那些抱怨缺少必要虚拟化技术资源和技术企业的 60%。

- **平台支持:** Xen 在新版本中支持更多的主机和客体工作环境，包括泛虚拟化技术、硬件辅助支持以及修改过或未修改过的客体操作系统；对 UNIX、Linux 和 Microsoft Windows 的特定支持；包括芯片组，如 x86、IA64 和 AMD、Fujitsu、IBM、Sun 等公司的 ARM，以及 x86/64 CPU 商家和 Intel 嵌入式的支持等。
- **可管理性:** EMA 在 2009 年度针对虚拟系统管理的研究结果表明：对于 83% 的企业来讲，在选择虚拟化技术过程中，管理是一项非常关键或者重要的因素。在对比 Xen 和 KVM 时，就可以看到 Xen 具有一个更加宽泛的第三方配给、备份、存储管理、P2V、容量规划、性能监控、进程自动化、安全和其它管理规则的社区，比如 Citrix、IBM、CA、Novell 或者 Platespin、Enomaly、Microsoft 以及 HP 等。
- **实施:** 无论 KVM 是“第一类”还是“第二类”，这都是语义上的概念。Xen 的运行和管理都是在较低级别的（Ring 0）。甚至对于新虚拟机创建，客体机器并不需要像 KVM 那样共享内存块、CPU 指令或者任何底层的 Linux 操作系统（虽然偶尔也会越权）。
- **KVM 不支持动态迁移:** 过去在证明 VMware ESX 比 Microsoft Hyper-V 优越的最重要的一个观点也同样适用于 Xen 和 KVM 的比较，但是这是一个很大的问题。与 KVM 不同，Xen 支持非中断的动态迁移，可以允许动态工作负载平衡和例行维护而几乎没有任何宕机时间。KVM 的固有属性决定了其肯定有宕机时间。
- **性能:** 大多数 Xen 和 KVM 性能基准的对比都表明 Xen 具有更好的处理性能（接近于本地处理），只有在磁盘 I/O 方面略逊于 KVM。进一步来讲，独立测试表明随着工作负载的增加 KVM 的性能逐渐下降。通常情况下，在试图支持四个以上的客体虚拟机时就会崩溃。Xen 支持一个客体虚拟机的数量呈线性增长，可以支持 30 个以上工作负载同时运行。

更加深入广泛的 Xen 与 KVM 对比也将会表明 Xen 在虚拟网络支持、虚拟存储支持、安全增强、高可用性、容错处理、电源管理、HPC 或者实时支持、虚拟 CPU 可扩展性、跨平台兼容性。虚拟机可移植性、虚拟应用设备市场以及已经建立起来的云服务社区等诸多方面都要比 KVM 优越。因此 KVM 只是在技术上比较与众不同，并且拥有几个不错的应用场景。但是作为企业级的服务器虚拟化技术，和 Xen 比起来，KVM 尚有较大差距。

Andi Mann 是企业管理协会 (EMA: Enterprise Management Associates) IT 分析部门中研究系统管理和存储管理的副总裁。Mann 在技术和管理领域有 20 多年的 IT 经验，对主机、中型机、服务器和台式机上企业形态和软件方面都有深刻了解。Mann 是 EMA 系统管理研究项目的主要负责人，他主要关注数据中心自动化和虚拟化。访问 EMA 的主页了解更多有关 Mann 的信息。

KVM 在 Linux 整合中优于 Xen

投稿人: Sander van Vugt

即使没有实施广泛的 Xen 和 KVM 性能基准研究，仍然有很多理由足以说明 Linux 领导者（诸如 Red Hat 和 Ubuntu）对待 KVM 态度的原因。其中最明显并且最重要的因素就是 KVM 是 Linux 内核的一部分，Xen 只是一个安装在 Linux 内核下层的一个产品而已。

为什么这一点如此重要呢？其之所以重要是因为在过去 Xen 工作环境的补丁包不能够和 Linux 内核兼容。但是如果实施 KVM 的话，这个问题就可以很容易地得到解决。另外一个选择 KVM 的原因是 KVM 在 Linux 内核内部部署，这样可以很容易控制虚拟化进程。

Xen 的拥护者们声称 KVM 不如 Xen 技术成熟，并且缺少某些关键特性，如动态迁移和泛虚拟化支持。确实如此，Xen 工作环境中的泛虚拟化技术可以使虚拟机的操作更加高效，因为泛虚拟化技术直接和硬件进行交互。然而使用泛虚拟化技术需要修改操作系统，默认 Windows 的安装并不支持泛虚拟化工作环境。至于动态迁移，KVM 也可以做到，只是需要安装正确的 KVM 版本而已。KVM 过去在动态迁移方面确实存在一个问题，但是现在已经解决了。

从另外一个角度来讲，KVM 更加灵活。由于操作系统直接和整合到 Linux 内核中的虚拟化管理程序交互，所以在任何场景下都可以直接和硬件进行交互，而不需要修改虚拟化的操作系统。这一点非常重要，因为对虚拟机运行来讲 KVM 可以是一个更快的解决方案。KVM 需要 Pacifica (AMD) 或者 Vanderpool (Intel) 虚拟化 CPU 这一事实在现在已经不能成为 KVM 发展的限制条件，因为当前大多数服务器 CPU 都有这些处理器。

支持不选择 KVM 虚拟化技术的一个可信原因是 Xen 相对来讲历史更久一点，产品更加成熟些。但是长远来看，Xen 将会使 Linux 内核的负担越来越重，因为 Xen 缺少很好的整合（并且以后也不会解决这个问题），尽管 Xen 的开发者们正在积极地解决这个整合问题。

最后一点是 KVM 是 Linux 内核的一部分，而 Xen 做到最好也无非是整合到 Linux 中。随着时间的推移，Red Hat（目前掌握 KVM 技术），作为 Linux 企业市场中份额最大的企业，将会使虚拟化技术的后来者——KVM 同 Xen 一样功能齐全。我相信未来虚拟化市场必定是 KVM 的。

(来源: TechTarget 中国)

KVM 再次引爆 Type 1 与 Type 2 hypervisor 战争

据 IBM 首席虚拟化架构师 Michael Day 所说，基于内核的虚拟机（KVM）口碑不佳更多是由于人们对它究竟是 [Type 1](#) 还是 Type 2 虚拟化平台存在误解。

争论由来已久，但是 Type 1 与 Type 2 服务器虚拟化平台之间的区别仍在不断讨论之中，尤其是提到 KVM 的时候，因为它不适合归入任何一个类型中。

Michael Day 在上周的 Linux 基金会合作峰会上就 KVM 神话做了演讲报告，他说：“围绕 KVM 的安全和性能，人们已经得出它究竟是 Type 1 还是 Type 2 的结论，一些 IBM 的用户已经看到并提及如下看法：如果 KVM 作为 Type 2 虚拟化平台运行在操作系统之上，性能将更难达到。他们同时暗示这样也更加不安全。”

KVM 历史性争论

KVM 是 Linux 内核中的虚拟化层。与所有的服务器虚拟化实现类似，它包括被称为虚拟机监视器的元素，该元素对工作负载及物理硬件资源和提供给应用程序的虚拟硬件之间的转化进行隔离。

Type 1 和 Type 2 虚拟化平台之间的区别取决于 VM 监视器和客户操作系统之间的转化发生的次数。对 Type 1 虚拟化平台也就是裸机虚拟化平台来说，转化只发生一次。反过来说，Type 2 虚拟化平台需要两次过程来游遍主机操作系统和 VM 监视器。

两种类型的虚拟化平台之间的区别可以追溯到一篇名为“可虚拟化的第三代架构在形式上的需求”的论文，这篇论文在 1974 年发表，作者是 Robert Goldberg 和 Gerald Popek。当今，VMware vSphere、微软 Hyper-V 以及 Xen 虚拟化平台（提供开源和 Citrix XenServer 产品）被很多人认为是 Type 1 虚拟化平台，而 OSX 的 Parallels、[VMware Workstation](#) 以及 Oracle VM VirtualBox，诸如此类产品通常被认为是 Type 2 虚拟化平台。

对 KVM 是 Type 1 还是 Type 2 虚拟化平台的混淆及争论的出现，部分是因为它是 Linux 操作系统的一部分。举例来说，在 2009 年，当时分析师 Andi Mann 就 [Xen](#) 和 KVM 的对峙写到：“KVM 是 Type 1 还是 Type 2 主要取决于语义。”

然而，Andi Mann 在那时强调：“Xen 在更低的层 (ring 0) 运行并被管理，即使是创建新虚拟机，用户之间也不共享内存块，CPU 指令或任何底层的（即使偶尔

享有特权) 的像 KVM 这样的 Linux 操作系统所共享 的资源。这意味着 KVM 遭受着性能、延迟、安全、可扩展性、隔离性以及其它问题。而这些问题并不会影响真正的裸机虚拟化平台。”

IBM 的 Michael Day 并不同意上述观点。在他看来, KVM 符合 Type 1 虚拟化平台的定义, 有两点原因: 第一, 它以内核模式在裸机上运行, 并且使用硬件虚拟化。第二, KVM 用户几乎花费他们所有的时间以直接执行模式运行, 这 是 Type 1 虚拟化平台的另一个评判标准, 这个标准最初由 Robert Goldberg 定义。

Michael Day 又说道: “有趣的是 Robert Goldberg 在他之后的论文中摒弃了 Type 1 和 Type 2 虚拟化平台之间的区别。现在区别再次被提及, 有很多博客对 Type 1 和 Type 2 虚拟化平台进行区分。”

现在 Type 2 虚拟化平台也不和之前的标准相匹配, 因为自他们的论文发表以来, 硬件和操作系统都已经改变了很多。

Type 1 还是 Type 2? 非当务之急

虚拟化的赞成者说, 当提及产品时, Type 1 和 Type 2 虚拟化平台之间的区别依旧明显。

Bob Plankers 是大学的一名虚拟化架构师, 他说: “重要的是我没有更多的工作要做。如果是 Type 2 虚拟化平台, 将给我带来很多的工作, 因为我不得不维护底层的操作系统以及虚拟化平台。如果是 Type 1 虚拟化平台, 所有的元素被封装在一起, 能够一起维护。”

然而, Bob Plankers 主要使用的虚拟化产品是 VMware vSphere, 当提及虚拟化管理和云计算的高级特性时, VMware vSphere 在几年前已经抢在了竞争对手, 包括 KVM 的前面。

Bob Plankers 说: “对我来说, 实质上是支持并兼容 KVM。每个人都了解 VMware, VMware 不遗余力地与所有人一同工作。它支持大量的硬件, 也有很多软件支持在 VMware 虚拟化之下运行……相反, KVM 没有如此成熟, 明显没有相同数量级的受支持硬件。”

IBM Linux 主管 Jean Staten 的一份声明中说到: “基于客户反馈, 对 Type 1 和 Type 2 的讨论不是是否采纳 KVM 的前提。对今天开始关注虚拟化数据中心的用户来讲, 采纳 KVM 是一个非常合乎逻辑的选择。然而, 已经锁定专有虚拟化平台

的用户 可能发现转换更加具有挑战性，因此他们可能需要时间把 KVM 融入到全部的数据中心战略中。

(来源: TechTarget 中国)

Red Hat vs. Ubuntu KVM 虚拟化：谁最好？

基于内核的虚拟机虚拟化技术是一项在开源社区内正在兴起的技术。

Canonical 公司的 Ubuntu 服务器版操作系统是第一个提供全功能的 KVM 虚拟化栈的主要 Linux 发行版，Red Hat 的企业版 Linux (RHEL: Red Hat Enterprise Linux) 一年之后随之发布。如果对这项虚拟化技术感兴趣的话，就需要权衡是选择 Red Hat 还是 Ubuntu KVM？

Red Hat 和 Ubuntu 操作系统

在回答这个问题之前，首先看一下这些 Linux 发行版。RHEL 是主要的企业 Linux 发行版，全球 500 强的企业都在使用该发行版作为其数据中心。RHEL 用来运行业务关键、高性能应用程序，其中包括如 Oracle 数据库和中间件产品以及 Red Hat 的 Jboss 生产线。

Ubuntu 服务器却与其它产品截然不同。Canonical 公司最近发布的 Ubuntu 服务器调查报告表明 Ubuntu 仍然在自定义组装的 PC 机中广泛使用——尤其是网页、邮件和数据库服务器中（如传统的 LAMP 栈）。另外，Ubuntu 也有达到企业级的潜能。然而问题就在于其管理能力是否能够使 Ubuntu KVM 满足当前企业级这一关键时期。

在 Red Hat 和 Ubuntu KVM 虚拟化的底层技术并没有区别。Red Hat 收购了 Qumranet，该公司的业务是开发 KVM，但是 KVM 本来就是一项开源技术。Red Hat 开发出来的所有功能迟早也都会出现在 Ubuntu 的产品中出现，反之亦然。

Red Hat 和 Ubuntu KVM 的实施

Red Hat 和 Ubuntu KVM 虚拟化在最佳应用场景中有所不同。Ubuntu 服务器中的基本主题是云计算，并且桉树系统 (Eucalyptus System) 支持的 Ubuntu 企业云已经整合在 9.04 版的操作系统中了。

对虚拟化有需求的客户并非经常需要云服务，因此并不是每一个用户都会选择 Ubuntu KVM。如果忽略云服务不看的话，Ubuntu KVM 只提供了一些诸如虚拟管理器和 virsh 的基本工具，这些工具都可以在 libvirt 库内使用。但是 Ubuntu KVM 可能不是企业级工作环境的最佳选择。

另一方面，Red Hat 为保证 KVM 的实施对 Red Hat 企业虚拟化服务器版以及针对服务器产品的 Red Hat 企业虚拟化（RHEV：Red Hat Enterprise Virtualization）管理器进行了进一步的开发。使用针对服务器的 RHEV 管理器来管理虚拟工作平台，并且可以其它管理平台竞争，其中包括 Citrix 系统公司的 XenCenter 和 VMware 公司的 vCenter 服务器。RHEV 可以提供高可用性、动态迁移、存储管理、系统调度以及其它 功能。

Red Hat 同时也非常积极地开发新的虚拟化工具和技术，比如 oVirt——针对虚拟工作环境基于网页的管理平台，libguestfs——虚拟机镜像的文件系统。正是归功于这些工具的出现以及服务器产品线的 RHEV，Red Hat 公司才得以在 KVM 虚拟化市场中占据主导地位。

尽管 Ubuntu KVM 的前景看起来非常不错，但由于上述优势的存在，Red Hat 仍然是必须认真考虑的唯一 KVM 虚拟化选择。

(来源: TechTarget 中国)

从 Xen 到 KVM: 弄清 RHEL 6 的变化

随着最新版操作系统的发布，红帽公司也放弃了以开源 Xen 为虚拟化平台的思路。Red Hat Enterprise Linux 6 开始支持 KVM (Kernel-based Virtual Machine) 作为 hypervisor。可能有人会惧怕迁移过程，实际上 Xen-to-KVM 迁移并不难。

确实，红帽的这种变化对于已经采用了 Xen 虚拟化平台的用户不应该成为困扰。该供应商依然支持采用 Xen 平台的 Red Hat Enterprise Linux (RHEL) 5 版本。同时对于那些需要迁移到 KVM 的用户，RHEL 6 中还包含了 Xen-to-KVM 迁移工具。

RHEL 6 的新虚拟化功能

Xen-to-KVM 迁移工具 virt-v2v，为 Xen 虚拟机迁移到 KVM 提供了简明易用的管理界面。该工具通过直接访问虚拟机文件系统并修改相关参数，实现从 Xen 或 VMware ESX 迁移到 KVM (Xen 和 KVM 镜像文件之间的差别其实很小)。Virt-v2v 在转换虚拟机的同时还同时加载 Virtio 驱动。

RHEL6 提供的另一个工具是 libguestfs，方便了对虚拟机磁盘镜像文件的访问和调整。

Libguestfs 本质上是一个允许管理员可以在虚拟机内直接挂载文件系统来修改客户参数的驱动。

RHEL 6 进一步增强了对虚拟化的支持。这些功能对于用户而言可能不太明显，但意义重大。KVM 的虚拟化仿真层在 RHEL 6 中不再那么明显，也就是可以做更多的直接管理动作。Virtio 工具也进行了修改，提高对硬件的利用效率。这些新功能使对虚拟机的访问更快，而且优化了内部资源调配。

RHEL 6 中最重要的一点变化是增加了高可用集群资源管理工具——Pacemaker，对于虚拟化而言很重要，因为它使虚拟机独立于底层硬件。结合 KVM 内置的在线迁移模块，高可用功能可以帮助管理员避免停机带来的影响。

Xen 到 KVM 的迁移并非所有的都改变了，毕竟在 RHEL 6 中虚拟机的管理都是一样的，使用了相同的基于 libvirt 的管理界面。在 RHEL6 中依然采用了 Virtual Machine Manager 和基于 shell 的 virsh 工具。

(来源: TechTarget 中国)

RHEL6 里 KVM 网络运行的两种方式

Red Hat Enterprise Linux ([RHEL](#)) 6 现在支持基于内核的虚拟机 (KVM) 管理程序。[KVM](#) 的抽象层——称为 libvirt——运行 KVM 网络，管理员能够在 libvirt 的配置文件中定制虚拟机连接并追踪网络设备。

在 libvirt 中运行 KVM 网络有两种方法：网络地址转换 (NAT) 和桥接网络。网络地址转换是默认方法，桥接网络仅对非常特殊的基础设施有意义。

对大多数管理员来说，最好从理解 KVM 网络在默认安装中如何工作开始。如果想创建定制安装，通过学习导航用于网络地址转换的 NAT 配置文件，同样可以熟悉相关设置。例如，你可以创建多个允许你改变不同虚拟机之间的网络流量的 NAT 接口。

NAT 配置文件

你可以在 `/usr/share/libvirt/network/default.xml` 文件中找到默认网络地址转换的 NAT 配置。下面是 `default.xml` 的一个例子：

```
[root@flo networks]# cat default.xml
<network>
<name>default</name>
<bridge name="virbr0" />
<forward/>
<ip address="192.168.122.1" netmask="255.255.255.0">
<dhcp>
<range start="192.168.122.2" end="192.168.122.254" />
</dhcp>
</ip>
</network>
```

NAT 配置文件首先命名网络设备。这有点令人困惑，但是默认的网络地址转换是一个 KVM 网络桥接设备。然后 NAT 配置定义了 IP 地址。KVM 网络桥接有它自己的 IP 地址和一系列相关的 IP 地址，这些 IP 地址可以使用 DHCP 分配给虚拟机。通过修改配置文件并重启网络或者通过使用 `virt-manager` 中的图形管理界面，KVM 的虚拟机管理器，你能改变这些 KVM 网络参数。

使用 `virsh net-list` 命令，你可以自动启动默认 NAT 接口。启动物理主机后，你将看到一个 `virbr0` 设备，当你启动虚拟机，他们将连接到这个设备并使用网络地址转换彼此连接并连接到主机。下面这个 NAT 配置显示两个虚拟机如何使它们的网卡连接到 `virbr0` 设备。

```
[root@flo networks]# brctl show
```

Bridge name↕	Bridge ID ↕	STPenabled↕	Interfaces↕
virbr0 ↕	8000.fe5400113597↕	yes↕	vnet0↕
↕	↕	↕	vnet1↕

每个虚拟机的配置文件同样表明了哪个网络连接虚拟机到主机。例如：

```
<interface type='network'>
<mac address='52:54:00:11:35:97' />
<source network='default' />
<model type='virtio' />
<address type='pci' domain='0x0000' bus='0x00' slot='0x03'
function='0x0' />
</interface>
```

在这个虚拟机配置中，虚拟机连接了主机的默认网络，但是你可以自由连接主机上的其他网卡，提供更加灵活的 KVM 网络。

为实现虚拟网络，主机自动建立 IP 转发，以及创建网络地址转换策略的 `iptables` 规则。

IP 转发在 `/proc/sys/net/ipv4/ip_forward` 文件中指定，必须包含值 1 以启用转发。`iptables` 规则被写入到主机的 NAT 表的路由前（Prerouting），路由后（Postrouting）和转发链。确保你不会意外改变他们，因为那将破坏到主机的 KVM 网络连接。

KVM 虚拟化：你用 RHEV 还是 RHEL？

Red Hat Enterprise Linux 与 Red Hat Enterprise Virtualization 都提供 [KVM](#) 虚拟化，但这两者在 KVM 管理、功能与实施中有重大区别。

RHEL 适合小型环境

Red Hat Enterprise Linux ([RHEL](#)) 是款普遍的 Linux 服务器产品，依赖 KVM 虚拟化。它由 Linux 内核与大量包组成，包括 Apache Web 服务器与 MySQL 数据库，以及一些 KVM 管理工具。使用 [RHEL 6](#) 可以安装并管理少量虚拟机，但不能交付最佳的性能与最优的 KVM 管理平台。当然，在小型环境中，RHEL 6 能满足你对开源虚拟化的所有要求。

为企业级环境准备的 KVM 虚拟化

如果虚拟化是你 IT 架构中的关键部分，那么你就不仅仅需要管理一些虚拟机了。对于企业级 KVM 虚拟化，你要的是轻松的 KVM 管理、高可用性、最佳性能与其他高级功能。Red Hat Enterprise Virtualization ([RHEV](#)) 包含 RHEV Manager (RHEV-M)，这个是集中的 KVM 管理平台，能同时管理物理与虚拟资源。

RHEV-M 能管理虚拟机与其磁盘镜像，安装 ISO，进行高可用性设置，创建虚拟机模板等，这些都能从图形 web 界面完成。也可使用 RHEV-M 管理两种类型的 hypervisor。RHEV 自身带有一个独立的裸机 hypervisor，基于 RHEL 与 KVM 虚拟化，作为托管的物理节点使用。另外，如果你想从 RHEV 管理运行在 RHEL 上的虚拟机，可注册 RHEL 服务器到 RHEV-M 控制台。

RHEV 3.0 还是 RHEL 6？

RHEL 6 就是一台 Linux 服务器，提供开源虚拟化选项，因此在大型环境就需要 RHEV 来实现 KVM 虚拟化。今年稍后要发布的 RHEV 3.0 有个重新设计的核，并且基于 Java。在 RHEV 3.0 中，你会发现它拥有竞争对手 [VMware](#) 或 [Citrix XenServer](#) 的所有功能。此外，RHEV 提供这些功能的价格低得多。

(来源: TechTarget 中国)

KVM 三大存储选项使用说明

KVM 的存储选项有多种，包括虚拟磁盘文件、基于文件系统的存储和基于设备的存储。

为实现 [KVM](#) (Kernel-based Virtual Machine) 存储管理，可以使用 LVM (Logical Volume Manager) 和创建存储池。当系统创建 KVM 虚拟机的时候，默认使用 [虚拟磁盘](#) 文件作为后端存储。安装后，虚拟机认为在使用真实的磁盘，但实际上看到的是用于模拟硬盘的虚拟磁盘文件。这一额外的文件系统层会降低系统速度。

当然，基于 [磁盘镜像](#) 的虚拟磁盘并非全无益处，磁盘文件系统可以很轻松地用于其它的 KVM 虚拟化宿主机。但是如果您希望优化 KVM 虚拟化性能，最好考虑其它的存储方式。

基于文件系统的 KVM 存储

在安装 KVM 宿主机时，可选文件系统为 dir (directory) 或 fs (formatted block storage) 作为初始 KVM 存储格式。默认选项为 dir, 用户指定本地文件系统中的某一个目录用于创建磁盘镜像文件。

fs 选项可以允许用户指定某个格式化文件系统的名称，把它作为专用的磁盘镜像文件存储。两种 [KVM](#) 存储选项之间最主要的区别在于:fs 文件系统不需要挂载到某个特定的分区。

两种选项所指定的文件系统，都可以是本地文件系统或位于 SAN 上某个物理宿主机上的网络文件系统。后者具备一定的优势，因为 SAN 可以很轻易地实现多个主机同时访问，而本地磁盘或文件系统则无法实现。

还有一种基于文件的磁盘存储方式是 netfs, 用户可以指定一个网络文件系统的名称，如 Samba。用这种方式作为 KVM 存储很方便，因为这样很容易访问到位于其它服务器上的文件系统，同时用户也可以通过多台宿主机访问磁盘文件。

所有的这些基于文件的 KVM 存储方式都有一个缺点：文件系统固有缺陷。因为虚拟机的磁盘文件不能直接读取或写入 KVM 存储设备，而是写入宿主机 OS 之上的文件系统。这也就意味着在访问和写入文件时中间增加了额外一层，这通常会降

低性能。所以，如果您希望寻找 KVM 虚拟化性能最优方案，最好考虑基于设备的存储。

基于设备的 KVM 存储

另外一种 KVM 存储的方式就是使用基于设备的方式。共支持四种不同的物理存储：磁盘、[iSCSI](#)、SCSI 和逻辑盘。磁盘方式指直接读写硬盘设备。iSCSI 和 SCSI 方式可选，取决于用户采取 SCSI 或 iSCSI 地址把磁盘设备连接。这种 KVM 存储方式的优势在于，磁盘的名称是固定的，而不需要取决于宿主机 OS 搜索到磁盘设备的顺序。

这种连接磁盘的方式也有缺点：灵活性不足。虚拟磁盘的大小很难改变，而且基于设备的 KVM 存储不支持快照。

如果要优化 KVM 存储的灵活性，可以使用 LVM (Logical Volume Manager)。LVM 的优势在于可以使用快照，而快照并不是 KVM 虚拟化自带的功能。

LVM 可以把所有存储放到一个卷组里，从而轻松创建一个逻辑卷。该卷组是物理磁盘设备的一个抽象，所以如果超出可用磁盘空间最大值，还可以向卷组中添加新的设备，从而极大简化了增加存储空间的过程，增加的空间在逻辑卷中直接使用。使用 LVM 使得磁盘空间分配更加灵活，而且增加和删除存储也更为容易。

最后，LVM 无论是在单宿主机或多宿主机环境中都可以很好工作。在多宿主机环境中，您可以在 SAN 上创建逻辑卷。如果使用 Cluster LVM，可以很容易的配置成多个主机同时访问某个逻辑卷。

使用 KVM 存储池

为简化 KVM 存储管理的目的，可以创建存储池。在宿主机上创建存储池，可以简化 KVM 存储设备的管理。采用存储池的方式还可以实现对提前预留的存储空间的分配。这种策略对于大型应用环境很有效，存储管理员和创建虚拟机的管理经常不是同一个人。这样，在创建首台虚拟机之前先完成 KVM 存储池的创建是很好的方法。

当您决定开始 KVM 虚拟化时，先在宿主机端创建一个 KVM 存储池，然后通过这个池提供 LVM 逻辑卷。对于使用 LVM 增加的[快照](#)功能，用户是不会感到后悔的，这种 KVM 存储方法提供了极大地灵活性。

如何调整 KSM 提升 KVM 性能？

虽然基于内核的虚拟机（[KVM](#)）被设计为自调优，但是可以调整一些参数使 KVM 主机性能更好。最重要的参数是内核同页合并（kernel samepage merging ,KSM），这一特性允许内核更有效地处理内存。KSM 允许 Linux 内核识别出包含相同内容的内存页，然后合并这些内存页，将数据整合在一个位置可以多次引用。

如果在主机上使用 KVM，通常会激活数个客户操作系统，而且这些操作系统经常运行相同的 OS，这意味着大量的内核页面被多次加载。通过应用 KSM，许多虚拟机可以使用相同数量的内存启动。事实上，KSM 允许虚拟机过度分配内存。但是使用 KSM 存在性能损失，在一般的环境中，性能损失大概是 10%，这也是在某些环境中关闭 KSM 的原因。

在 RHEL 6 和 Fedora 14 中，KSM 默认是打开的。KSM 通过两个服务：ksmd 和 ksmtuned 实现，这两个服务在系统初始化时自动启动。管理员应该判断他们的环境并决定保持 KSM 处于运行状态还是关闭它。

决定使用 KSM 还是关闭

如果目标是运行尽可能多的虚拟机，而且性能不是问题，应该保持 KSM 处于运行状态。例如 KSM 允许运行 30 个虚拟机的主机上运行 40 个虚拟机，这意味着最大化硬件使用效率。但是，如果服务器在运行相对较少的虚拟机并且性能是个问题时，那么应该关闭 KSM。

对任何系统来说，最佳选择将取决于创建虚拟环境时的内存估算。如果在虚拟主机中有足够的物理内存，在没有开启 KSM 时就能够满足虚拟机的内存需求，那么最好关闭 KSM。使用 `chkconfig ksmd off`, `chkconfig ksmtuned off`, `service ksmd off` and `service ksmtuned off` 命令关闭 KSM。但是如果主机内存紧张，那么最好保持 KSM 处于运行状态。

优化 KSM 达到最佳性能

使用 KSM 时，可以优化一些参数以达到最佳性能。这些参数位于一个小的配置文件 `/etc/ksmtuned.conf` 中：

```
#优化 KSM 的配置文件
#优化调整之间应休眠多长时间
# KSM_MONITOR_INTERVAL=60
#在扫描 16Gb 服务器之间 ksm 休眠的毫秒数
#内存较小的服务器休眠的时间更长，内存较大的服务器休眠时间更短。
# KSM_SLEEP_MSEC=10
# KSM_NPAGES_BOOST=300
# KSM_NPAGES_DECAY=-50
# KSM_NPAGES_MIN=64
# KSM_NPAGES_MAX=1250
# KSM_THRES_COEF=20
# KSM_THRES_CONST=2048
#如果你想获取优化 KSM 的调试信息，取消以下注释
# LOGFILE=/var/log/ksmtuned
# DEBUG=1
```

配置文件中最重要参数是 KSM_SLEEP_MSEC。Fedora 14 使用的默认值是大型服务器的设置值。当在主机上运行较少的虚拟机，使用 KSM 时最好让主机休眠更长的时间。例如，尝试设置 KSM_SLEEP_MSEC=50，然后测试对虚拟机的影响。

(来源: TechTarget 中国)

通过 virsh 执行 KVM 流程化管理

在进行 KVM 虚拟化管理时，virsh 的学习曲线非常陡峭。但是另一种可选方式 virt-manager 的缺陷还很明显。

virt-manager 需要在系统内运行，这会浪费资源并为别有目的的人提供额外的进攻点。virt-manager 的另一大缺点是需要通过图形工作站运行远程接入的方式，才能对服务器上的虚拟机进行管理。

而 virsh 则可以通过命令行执行各种各样的管理任务。本文例举了一些最重要的 virsh 命令。

Virsh list

对 virsh 初学者而言，可以先列举出所有可用的虚拟机。为达到该目的，输入 virsh list 命令，下面是该命令输出结果的举例：

```
sander@boston:~$ sudo virsh list
Id Name State
```

```
-----
1 sles10sp3 running
```

不过，这个命令无法提供更多细节内容。如果需要更详细，输入 virsh dominfo 命令。如下关于名为 sles10sp3 主机的信息就是通过 virsh dominfo 命令获得的：

```
sander@boston:~$ sudo virsh dominfo sles10sp3
Id: 1
Name: sles10sp3
UUID: 0ecd18d7-dec7-0668-9fc4-9bf2fbale1c7
OS Type: hvm
State: running
CPU(s): 1
CPU time: 172.4s
Max memory: 524288 kB
Used memory: 524288 kB
```

```
Autostart: disable
Security model: apparmor
Security DOI: 0
Security label: libvirt-0ecd18d7-dec7-0668-9fc4-9bf2fbale1c7 (enforcing)
```

virsh 完成虚拟机关机和重启

关闭虚拟机是最虚拟化后常见的任务。在 KVM 环境中，您可以使用如下的命令来完成该动作：

- `virsh shutdown`：这个命令会停止所有的活动进程然后关闭虚拟机。
- `virsh destroy`：该命令会强制关闭虚拟机，可能因此数据丢失。执行效果类似于拔掉物理机的电源。

如果要重新启用离线的虚拟机，使用 `virsh start` 命令，后面加上虚拟机的名称。

通过 virsh 完成虚拟机的待机、唤醒和恢复

停止虚拟机有些时候不是最理想方式。有时，您可能希望冻结虚拟机，稍后再快速恢复到当前状态。这种方式特别适合于那些虚拟机上打开多个应用或窗口而且需要依次进行处理的情况。

在这种情形下，输入 `virsh suspend` 后加虚拟机名称。稍后，如果需要让虚拟机在线，使用 `virsh resume` 命令。

对于子虚拟机的备份和恢复也非常地重要。为了保存虚拟机的现有状态，使用 `virsh save` 命令。有了状态文件，用 `virsh restore` 命令可以恢复虚拟机。假设用户配置错误，这些命令还可以把虚拟机恢复到之前的工作状态。

学习 virsh 命令更多相关内容

上面的例子只是为了说明 `virsh` 的可用性和多功能性。如果希望了解更多 `virsh` 内容，输入 `virsh help`，可以列举出所有的 `virsh` 命令。

假设您从 `help list` 中发现了 `virsh setmem` 命令，希望可以了解更多。输入 `virsh help setmem`，下面是输出结果：

```
sander@boston:~$ sudo virsh help setmem
```

NAME

setmem - change memory allocation

SYNOPSIS/p>

setmem domain kilobytes

DESCRIPTION

Change the current memory allocation in the guest domain.

OPTIONS

domain domain name, id or uuid

kilobytes number of kilobytes of memory

通过检索命令列表，可以对 virsh 的功能有全面的了解。不过，对于那些复杂的命令，需要一些外部资源的协助来加深理解。

(来源: TechTarget 中国)