

基于 KVM 虚拟化平台的构建研究

李晓佳, 董延华, 吴佳怡

(吉林师范大学 计算机学院, 吉林 四平 136000)

摘要: 云计算作为未来网络技术发展的方向, 可以快速构建各类高效的信息化系统, 其实现核心虚拟化技术也得到了用户广泛的重视。KVM 是一种高效的虚拟化技术, 提供了一种基于 Linux 内核的虚拟化解决方案, 通过结合 QEMU 提供设备虚拟化, 提高了资源分配的效率和灵活性, 从而确保了虚拟化平台的高效性和稳定性。本文从 KVM 虚拟化平台的构建入手, 对 KVM 虚拟化技术的优势进行分析, 具体阐述了 KVM 虚拟化的部署方案, 为集群的创建提供了硬件支撑。

关键词: 虚拟化技术; Linux 内核; KVM 虚拟化平台

Research on the construction of KVM based virtualization platform

LI Xiaojia, DONG Yanhua, WU Jiayi

(College of Computer Science, Jilin Normal University, Siping Jilin 136000, China)

[Abstract] As the direction of future network development technology, cloud computing can rapidly construct various kinds of efficient information systems, and its core virtualization technology has also been widely valued by users. KVM is an efficient virtualization technology that provides a virtualization solution based on the Linux kernel. By combining with QEMU to provide device virtualization, it improves the efficiency and flexibility of resource allocation, thus ensuring the efficiency and stability of the virtualization platform. This paper starts with the construction of KVM virtualization platform, analyzes the advantages of KVM virtualization technology, and elaborates the deployment scheme of KVM virtualization in detail, which provides hardware support for cluster creation.

[Key words] virtualization technology; Linux kernel; KVM virtualization platform

0 引言

随着云计算概念的兴起, 虚拟化技术也已逐渐引起学界关注, 这是云计算架构中的核心技术。研究中是用软件的方法重新划分和定义 IT 资源, 打破实体结构间的不可分割的障碍, 实现对 IT 资源的动态分配、灵活调度, 简化对资源以及对资源管理的访问, 从而提高资源的利用效率。

1 KVM 虚拟化技术

主流的虚拟化技术包括 KVM (Kernel-based Virtual Machine)、Xen、VMM 等, 其中, KVM 是集成到 Linux 内核的虚拟机管理程序 (Hypervisor), 是 X86 架构且硬件支持虚拟化技术的全虚拟化解决方案。KVM 技术核心是实现了内核中对处理器虚拟化特性的支持, 但缺乏设备虚拟化以及相应的用户空间管理虚拟机的工具^[1]。相比较而言, XEN 属于外部的 Hypervisor, 能够控制虚拟机, 并给多个客户机分配资源, 但缺点是更新 XEN 版本时, 需要重新编译整个内核, 容易造成系统崩溃。相比之下, KVM 的结构更精简, 代码量更少。

2 KVM 与 QEMU、Libvirt 组件的关系

作为 Linux 内核驱动模块的一部分, KVM 可以用 Modprobe 模块支持 IntelVT 或 AMD-V 虚拟化, 并通过 /dev/kvm 暴露程序接口, 用户态程序可以通过 /ioctl 函数来访问这个接口, 使 CPU 的相关指令部分转交给内核来处理^[2]。

2.1 KVM 与 QEMU 的关系

QEMU 是一个独立的虚拟化解决方案, 包括独立的处理器虚拟化、内存虚拟化以及 I/O 设备虚拟化, 其本身可以不依赖 KVM 而存在, 但由于 KVM 无法独立完成用户与内核模块间的任务交互, 需要借助 QEMU 作为用户空间的管理工具, 而 QEMU 同样需要通过 KVM 提高硬件虚拟化的性能。

2.2 KVM 与 Libvirt 的关系

Libvirt 是面向驱动的架构设计, 由 API 库、Libvirtd 服务以及 virsh 命令行管理工具组成, 可以通过调用闲置的 API 库实现对各种类型虚拟机的管理^[3]。Libvirt 提供了多种语言的编程接口, 可以直接通过编程调用 Libvirt 提供的对外接口实现对虚

拟机的操作,不同的虚拟机提供不同的驱动,从而实现相互间不受影响,有利于后期的研发拓展。

3 KVM 虚拟机的实现

KVM 的获得方式有多种,研究常用 yum 命令直接进行安装,操作的前提是宿主机(服务器)需要获得支持 KVM 的权限。对此可做阐释论述如下。

3.1 开启虚拟化服务

(1)安装虚拟化软件配置信息如下:

```
#yum groupinstall Virtual *
```

(2)检查 KVM 是否开启成功。配置代码为:#

```
lsmod | grep KVM
```

运行后,出现如图 1 所示的反馈结果即支持服务器虚拟化操作。

```
kvm_intel          54285  0
kvm                333172  1 kvm_intel
```

图 1 查看 KVM 环境配置

Fig. 1 Check the KVM environment configuration

3.2 KVM 组件安装及网络配置

(1)启动 libvirt 脚本,此时将用到的配置代码可写为:#Service libvirtd start

(2)配置网卡的桥接信息。在安装和启动 libvirtd 服务后,libvirtd 在服务器(宿主机)上自动生成一个 virbr0(virtual network switch),连接在宿主机上的虚拟机会默认通过 virbr0 连接起来。但由于设计采用的是 NAT 网络模式,在连接多台虚拟机时会造成 IP 地址的浪费,因此在操作过程中,一般将 virbr0 予以关闭,再使用 br0 进行桥接设置。具体配置如下:

```
# virsh net-list
Name      State      Autostart
default   active     yes
# virsh net-destroy default
Network default destroyed
# virsh net-undefine default
Network default has been undefined
# service libvirtd restart
Stopping libvirtd daemon:[ OK ]
Starting libvirtd daemon:[ OK ]
```

3.3 创建虚拟服务器

(1)通过 QEMU-img 创建 8 G 的映像文件。配置代码如下:

```
#qemu-img create -f raw /share/class1.raw 8G
```

(2)安装 CentOS 6.6 版本的操作系统,配置如

```
下:#virt-install --virt-type kvm --name class1
--ram 1024 --cdrom=/ISO/CentOS-6.6-i386-
bin-DVD1.iso --disk path=/share/class1.raw --
-network network=default --graphics vnc,listen=
0.0.0.0--noautoconsole
```

3.4 开启远程桌面控制

Linux 中对虚拟桌面的连接共有 4 种方式,对此可得研究分述如下:

(1)宿主机(服务器)上使用 SSH 命令连接虚拟机文本界面。

(2)宿主机(服务器)上使用 virt-manager 命令进入虚拟机图形管理界面。

(3)宿主机(服务器)上使用 virsh 命令的子命令 concole 通过串口连接虚拟机。

(4)远程计算机通过虚拟计算机网络(Virtual Network Computing,VNC)连接虚拟机图形界面。

实验中采用 VNC 进行远程桌面的管理,具体配置如下:

(1)安装 VNC 服务包。对应代码为:

```
rpm -ivh /mnt/Server/vnc-server-4.1.2-9.
el5.x86_64.rpm
```

(2)配置虚拟机监听。对应代码为:

```
# vi /etc/libvirt/qemu.conf
vnc_listen = "0.0.0.0"
vnc_password = "666666"
```

(3)配置 VNC 图形桌面为 KDE。对应代码为:

```
# vi /root/.VNC/xstartup
Unset SESSION-MANAGER
Exec /etc/X11/xinit/xinitrc
```

4 结束语

作为当前应用最广泛的虚拟化技术之一,KVM 虚拟化为云计算的进一步发展和应用奠定了坚实的基础,如何提高虚拟化平台的性能是构建云平台的关键,本文从分析 KVM 虚拟化平台的构成入手,详细讨论了平台的实现方法,未来将深入研究基于 KVM 虚拟化的批量克隆技术,从而提高虚拟化集群搭建的效率。

参考文献

- [1] 彭天伟. KVM 虚拟化技术在云平台中的应用[J]. 电脑知识与技术,2015,11(14):32-33.
- [2] 彭晓平,张雪坚,黄波. 基于 KVM 的虚拟化技术研究[J]. 中国新通信,2017(20):77-80.
- [3] 王伟. 云计算原理与实践[M]. 北京:人民邮电出版社,2018.