**KVM基础实验**

* 实验目的：

学习并实践KVM虚拟化技术的基本使用

* 实验背景知识

KVM是Linux内核自带的虚拟化模块，和用户态的QEMU配合构成虚拟化平台。KVM利用了硬件辅助虚拟化技术，属于全虚拟化，虚拟机操作系统无需任何修改就可以直接安装。

Libvirt是一套免费、开源的支持Linux下主流虚拟化工具的函数库，为KVM、XEN等虚拟化平台提供了一套统一、方便、可靠的编程接口。Virsh是基于Libvirt的用于管理虚拟化平台的字符界面工具，通常由Libvirt安装包自带；Virt-manager同样是基于Libvirt的管理工具，但提供了图形化界面，方面操作。

* 实验步骤：

1. 找到一台支持Intel VT技术的物理机器，在这台机器上安装任意Linux系统，在BIOS中确认Intel VT技术已经打开。注意：不推荐使用VMware中安装的Linux，因为涉及到嵌套虚拟化。
2. 调研QEMU对KVM的作用，它们是如何相互配合的？下载Qemu代码并编译安装到系统。
3. 下载Libvirt以及Virt-manager代码并编译安装到系统。
4. 使用Virsh命令或者Virt-manager成功创建两台虚拟机，其中一台为Windows虚拟机，另一台为Linux虚拟机。
5. 调研虚拟机可以使用的虚拟网卡有哪几种类型，区别是什么？查看所安装的两台虚拟机使用的是哪种类型的虚拟网卡，以及它们之间的网络是否畅通。

* 参考资料

KVM <http://www.linux-kvm.org/page/Main_Page>

QEMU <http://wiki.qemu.org/Main_Page>

QEMU Install <http://wiki.qemu.org/Hosts/Linux>

Libvirt <http://libvirt.org/>

Libvirt Install <http://libvirt.org/compiling.html>

Virt-manager Install <http://ask.xmodulo.com/compile-virt-manager-debian-ubuntu.html>

虚拟机安装 <https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/6/html/Virtualization_Host_Configuration_and_Guest_Installation_Guide/chap-Virtualization_Host_Configuration_and_Guest_Installation_Guide-Guest_Installation_Virt_Manager-Creating_guests_with_virt_manager.html>

* 要求与提示

重要实验步骤需给出截图。

编译安装QEMU、Libvirt、Virt-manager时可能会缺少依赖包，可根据报错提示逐一安装依赖包。

如果编译安装有困难，可以选择使用系统自带的包管理工具进行安装，但会酌情减分。

def createConnection():

conn=libvirt.openReadOnly(None)

if conn==None:

print 'Failed to open connection to QEMU/KVM'

sys.exit(1)

else:

print '-----Connection is created successfully-----'

return conn

if \_\_name\_\_=='\_\_main\_\_':

name1= "centos7.0”

name2="notExist"

print "---Get domain info via libvirt python API---"

conn=createConnection()

getDomInfoByName(conn,name1)

getDomInfoByName(conn,name2)

closeConnection(conn)