**课程大作业三**

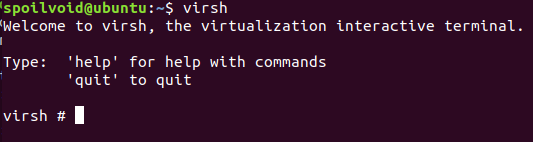
叶增渝 519030910168

P.S.本来是想要提交相关源文件的，但是发现提交文件类型仅为pdf，故将相关源码贴在文末

1.配置环境

（1）首先我们需要下载libvirt环境与对应的Python API

首先键入sudo apt-get install libvirt-dev libvirt-daemon libvirt-clients命令安装基本virsh命令。安装完成后我们键入virsh便可进入virsh命令行

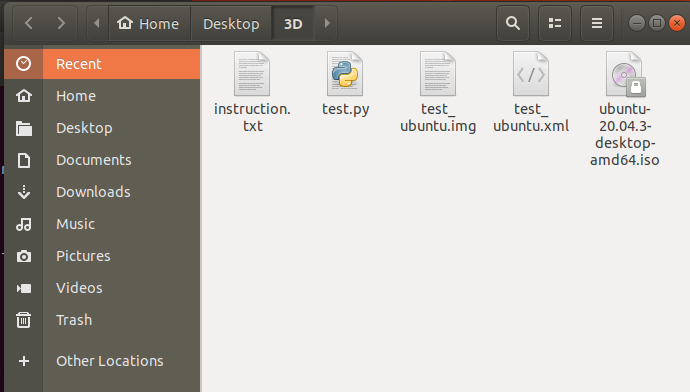


再使用sudo apt-get install python-libvirt命令安装Python API，以便后续Python脚本的运行

（2）配置虚拟机对应xml文件（源码见文末）

对于之前使用qemu创建的虚拟机，没有相关的xml文件，也就无法被virsh相关命令发现，所以第一步是在qemu虚拟机文件夹下创建xml配置文件并进行修改(由于之前已经安装系统了，所以<boot dev='hd'/>需要设置为hd防止进入时首先进入安装界面)

Tips:我在这里选择使用Vncviewer连接重新安装了qemu虚拟机，所以与作业2显示的img文件名不相同

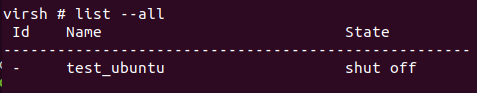


（3）利用virsh命令行工具将其纳入libvirt管理下

首先在virsh下使用define test\_ubuntu.xml命令



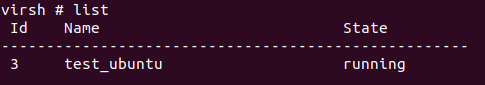
然后我们便可以在list –all命令下查看当前管理下的虚拟机



此时test\_ubuntu仍然处于关闭状态，我们使用start test\_ubuntu命令开启虚拟机



（在这之后此时我还使用了vncviewer使用GUI界面打开了虚拟机，但没有后续操作，我认为这是不必要的，故不加入步骤中）

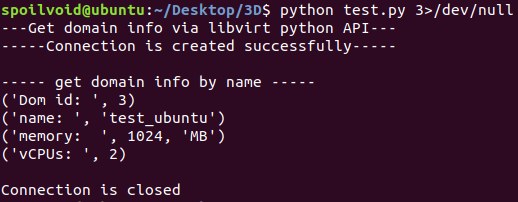


此时我们仅仅通过list命令就可以看到该虚拟机的name与id了

2.编写test.py脚本文件获取需要数据（源码见文末）

我们使用python libvirt-test.py 2>/dev/null命令执行Python脚本

这里的实现原理：通过自己设定好的虚拟机名字使用conn.lookupByName(name)进行查找，查询结果如下，观察结果与在xml中文件中设置的内容，不难发现我们成功获取了虚拟机的ID为3、name为”test\_ubuntu”、最大内存大小为1024MB与vCPU个数为2，任务完成。



test\_ubuntu.xml文件源码：

<domain type='kvm'>

<name>test\_ubuntu</name> //虚拟机名称

<memory>1048576</memory> //最大内存，单位k

<currentMemory>1048576</currentMemory> //可用内存，单位k

<vcpu>2</vcpu> //虚拟cpu个数

<os>

<type arch='x86\_64' machine='pc'>hvm</type>

<boot dev='hd'/> //光盘启动

</os>

<features>

<acpi/>

<apic/>

<pae/>

</features>

<clock offset='localtime'/>

<on\_poweroff>destroy</on\_poweroff>

<on\_reboot>restart</on\_reboot>

<on\_crash>destroy</on\_crash>

<devices>

<emulator>/usr/bin/kvm</emulator>

<disk type='file' device='disk'>

<driver name='qemu' type='qcow2'/>

<source file='/home/spoilvoid/Desktop/3D/test\_ubuntu.img'/> //目的镜像路径

<target dev='hda' bus='ide'/>

</disk>

<disk type='file' device='cdrom'>

<source file='/home/spoilvoid/Desktop/3D/ubuntu-20.04.3-desktop-amd64.iso'/> //光盘镜像路径

<target dev='hdb' bus='ide'/>

</disk>

<interface type='bridge'> //虚拟机网络连接方式

<source bridge='virbr0'/> //当前主机网桥的名称

<mac address="00:16:3e:5d:aa:a8"/> //为虚拟机分配mac地址，务必唯一，否则dhcp获得同样ip,引起冲突

</interface>

<input type='mouse' bus='ps2'/>

<graphics type='vnc' port='19' autoport='yes' listen = '0.0.0.0' keymap='en-us'/>//vnc方式登录，端口号自动分配，自动加1，可以通过virsh vncdisplay来查询

</devices>

</domain>

test.py文件源码：

#!/usr/bin/python

import libvirt

import sys

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

def closeConnection(conn):

print('')

try:

conn.close()

except:

print('Failed to close the connection')

return 1

print('Connection is closed')

def getDomInfoByName(conn, name):

print('')

print('----- get domain info by name -----')

try:

myDom = conn.lookupByName(name)

except:

print('Failed to find the domain with name %s',name)

return 1

print("Dom id: ", myDom.ID())

print("name: ", myDom.name())

print("memory: ",int(myDom.maxMemory()/1024), "MB")

print("vCPUs: ", myDom.maxVcpus())

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

name1 = "test\_ubuntu"

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

conn = createConnection()

getDomInfoByName(conn, name1)

closeConnection(conn)