How to install vm using KVM

1. Prepare net-work
   1. check exist bridge

# brctl show

Example output:

bridge name     bridge id               STP enabled     interfaces

virbr1          8000.5254000fed99       yes             virbr1-nic

                                                        vnet1

virbr2          8000.5254000fed99       yes             virbr2-nic

                                                        vnet0

virbr1&virbr2 had been used, so next step we can use virbr3,virbr4,virbr5. Otherwise we can use start from virb1.

* 1. Prepare network xml file

The path doesn’t matter but you should remember it. Recommend is /var/tong

#vi ne-om.xml

Content as below:

1. dhcp mode

<network>

  <name>ne-om</name>     //network name

  <uuid>bd2ddbf7-4c78-42d6-bf38-daa45ef8a060</uuid>  //define uui, must be unique, can use uuidgen to generate a new one

  <forward mode='nat'/>

  <bridge name='virbr3' stp='on' delay='0' />   //virtual bridge device

  <mac address='52:54:00:0F:ED:99'/>       //must be different with other within this KVM

  <ip address='192.168.10.1' netmask='255.255.255.0'>

    <dhcp>

      <range start='192.168.10.2' end='192.168.10.254' />

    </dhcp>

  </ip>

</network>

1. static ip mode

<network>

  <name>ne-om</name>     //network name

  <uuid>bd2ddbf7-4c78-42d6-bf38-daa45ef8a060</uuid>  //define uui, must be unique, can use uuidgen to generate a new one

  <forward mode='nat'/>

  <bridge name='virbr3' stp='on' delay='0' />   //virtual bridge device

  <mac address='52:54:00:0F:ED:99'/>       //must be different with other within this KVM

 <ip address='192.168.123.1' netmask='255.255.255.0'>

    <dhcp>

      <host mac='52:54:00:5b:fc:be' ip='192.168.123.3' />

    </dhcp>

  </ip>

</network>

* 1. check network

#virsh net-list --all

* 1. define network bridge

#virsh net-define ne-om.xml

* 1. check network, the define network should be inactive.

#virsh net-list --all

Name State Autostart Persistent

--------------------------------------------------

ne-oam inactive no yes

* 1. Active network:

#virsh net-start ne-om

# virsh net-list --all

Name State Autostart Persistent

--------------------------------------------------

default active yes yes

ne-om active no yes

* 1. Check ifconfig, it should has below bridge:

virbr3 Link encap:Ethernet HWaddr 52:54:00:0F:ED:80

inet addr:192.168.11.1 Bcast:192.168.11.255 Mask:255.255.255.0

UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1

RX packets:0 errors:0 dropped:0 overruns:0 frame:0

TX packets:0 errors:0 dropped:0 overruns:0 carrier:0

collisions:0 txqueuelen:0

RX bytes:0 (0.0 b) TX bytes:0 (0.0 b)

1. Prepare ISO file

Upload rhel-server-6.5-x86\_64-dvd.iso to /var/tong

Note: The path doesn’t matter but you should remember it.

1. Create disk

#cd /var/tong

#qemu-img create -f qcow2 DMF.img 50G

#qemu-img create -f qcow2 DMF-GFS.img 250G

1. Create img

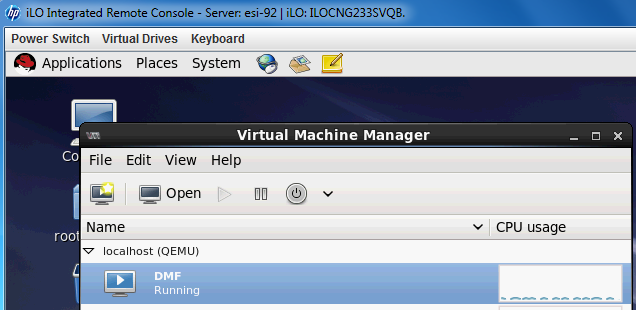
# virt-install --name DMF --hvm --ram 32768 --vcpus 6 --disk path=/var/tong/DMF.img,size=10,bus=virtio,format=qcow2 --network network:ne-om --accelerate --vnc --vncport=5920 --cdrom /var/tong/ rhel-server-6.5-x86\_64-dvd.iso --boot cdrom

1. open vm

There are several methods to open VM

* 1. using host console

If host machine OS had installed desktop , Select **Applications**->**Virtual Machine Manager**

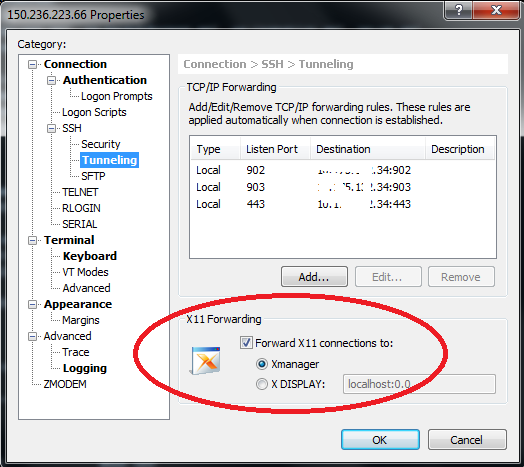


Select DMF, click **Open** button

* 1. Xshell4 and Xmanager

1. Install Xshell4 and Xmanager
2. In Xshell4 Tunneling

Open “X11 Forwarding”, and choose “Xmanager”



1. Open Xmanager
2. Open vm using virt-viewer

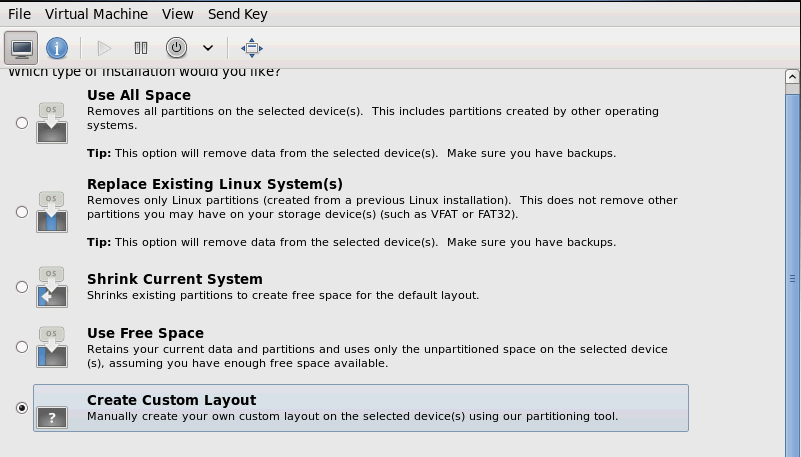
# virt-viewer {vm name}

For example:

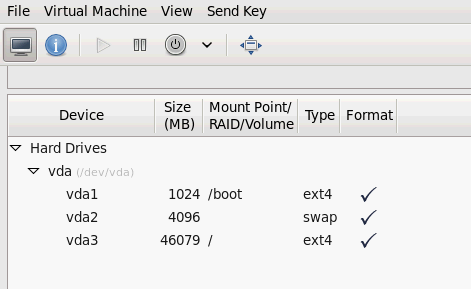
# virt-viewer DMF

1. Install VM

After open the VM, Keep press **next** to until to below step:



Select “Create Custom Layout” and create disk as below:



Keep press **next** until OS install finished.

1. Login to OS and enable network

#ifconfig eth0 up

#service network restart

#ifconfig

Note: if the network is not ok, please check /etc/sysconfig/network-scripts/ifcfg-eth0, the content looks like as below

#vi /etc/sysconfig/network-scripts/ifcfg-eth0

DEVICE=eth0

HWADDR=52:54:00:5b:fc:bf

TYPE=Ethernet

ONBOOT=yes

NM\_CONTROLLED=yes

BOOTPROTO=dhcp

1. Mount GFS disk

#vi /etc/libvirt/qemu/DMF.xml

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

<driver name='qemu' type='qcow2' cache='none'/>

<source file='/var/tong/DMF-GFS.img'/>

<target dev='vdb' bus='virtio'/>

<address type='pci' domain='0x0000' bus='0x00' slot='0x04' function='0x0'/>

</disk>

Note: Each device include should have different slot.

1. Mount other network
   1. Prepare network xml file

#vi /var/tong/ne-ingestion.xml

#vi /var/tong/ne-bmcbdcAccess.xml

The content refer to ne-om.xml

* 1. Define network

#cd /var/tong

#virsh net-define ne-ingestion.xml

#virsh net-define ne-bmcbdcAccess.xml

* 1. check network, the define network should be inactive.

#virsh net-list --all

Name State Autostart Persistent

--------------------------------------------------

ne-bmcbdcAccess inactive no yes

ne-ingestion inactive no yes

ne-oam active no yes

* 1. Active network

#virsh net-start ne-ingestion

#virsh net-start ne-bmcbdcAccess

#virsh net-list --all

Name State Autostart Persistent

--------------------------------------------------

default active yes yes

ne-bmcbdcAccess active no yes

ne-ingestion active no yes

ne-om active no yes

* 1. Add network to VM

#vi /etc/libvirt/qemu/DMF.xml

Add below two network to DMF.xml

<interface type='network'>

<mac address='52:54:00:5b:fc:bf'/>

<source network='ne-ingestion'/>

<address type='pci' domain='0x0000' bus='0x00' slot='0x07' function='0x0'/>

</interface>

<interface type='network'>

<mac address='52:54:00:5b:fc:bd'/>

<source network='ne-bmcbdcAccess'/>

<address type='pci' domain='0x0000' bus='0x00' slot='0x08' function='0x0'/>

</interface>

* 1. Restart vm

#virsh list --all

#virsh shutdown DMF

#virsh start DMF

* 1. Login to OS and enable the two network

#ifconfig eth1 up

#ifconfig eth2 up

#service network restart

#ifconfig

1. Change vm boot

#vi /etc/libvirt/qemu/DMF.xml

Change:

<os>

….

<boot dev='cdrom'/>

</os>

To:

<os>

…

<boot dev='hd'/>

</os>

1. Other useful command
2. Check img info

#qemu-img info <img file path>