

Autor: Iván Matarrubias Filgueira	Asunto: KVM/LIBVIRT
	Fecha: 6 de febrero de 2017



Manual de instalación KVM/LIBVIRT	
Fecha: 6 de febrero de 2017	Número de revisión: Versión 1, documento inicial
Objeto del documento: Manual de instalación y configuración	

Índice de contenido

1 Introducción.....	2
2 Primeros Pasos.....	3
3 Instalación.....	3
4 Images.....	3
5 Creación de máquinas virtuales.....	4
6 Configuración de red.....	6
7 Máquina de gestión.....	7
8 Administración de máquinas virtuales qemu/KVM.....	8

Autor: Iván Matarrubias Filgueira	Asunto: KVM/LIBVIRT
	Fecha: 6 de febrero de 2017



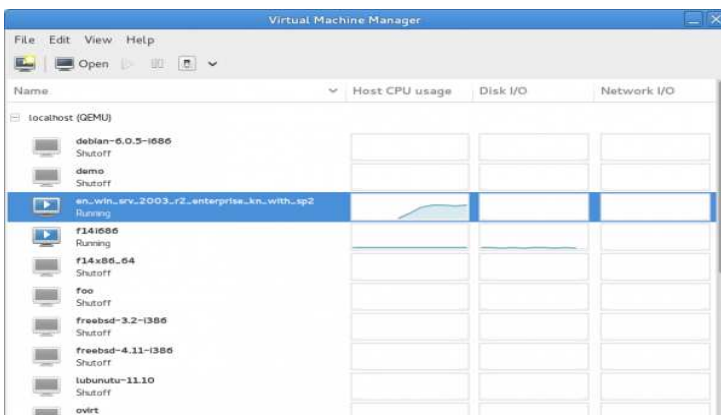
1 Introducción



[libvirt](#) es una API de código abierto, es una herramienta para la administración de máquinas virtuales, soporta los siguientes hypervisores **KVM/Qemu**, Xen, LXC, OpenVZ, VirtualBox, Hyper-V, y de VMware soporta ESX y Workstation.



KVM (Kernel based Virtual Machine) es un módulo del kernel que permite utilizar las extensiones de virtualización de los procesadores que en casi todos los procesadores actuales ya sea Intel (**vmx**) o AMD (**svm**) esta soportado tanto para arquitectura x86 como x86_64.



[virt-manager](#) es una aplicación de escritorio para administrar máquinas virtuales por medio de libvirt, su principal foco es para máquinas virtuales en KVM, pero también soporta Xen y LXC, su interfaz permite ver un resumen de las máquinas y una vista rápida de los recursos utilizados, tiene un asistente (wizard) para crear Máquinas Virtuales, tiene un cliente VNC y SPICE para tener acceso a la consola gráfica de las máquinas virtuales.

Autor: Iván Matarrubias Filgueira	Asunto: KVM/LIBVIRT
	Fecha: 6 de febrero de 2017



2 Primeros Pasos

Primero hay que verificar que nuestro procesador soporta las extensiones de virtualización, si la máquina donde vamos a probar tiene un procesador fabricado aproximadamente después del 2006 no deberíamos tener problema, siempre y cuando no sean de la gama baja, Intel Atom, Intel Celeron.

#grep --color -e svm -e vmx /proc/cpuinfo

```
root@viva1t-gemu89:~/scripts# grep --color -e svm -e vmx /proc/cpuinfo
flags       : fpu_vme_de_pse_tsc_msr_pae_mce_cx8_apic_sep_mtrr_pge_mca_cmov_pat_pse36_clflush_dts_acpi_mmx_fxsr_sse_sse2_ss_ht_tm_pbe_syscall_nx_lm_constant_tsc_arch_perfmon_pebs_bts_rep_good_nopl_aperfperf_pni_dtes64_monit
or_ds_cpl_vmx_est_tm2_ssse3_cx16_xtpr_pcdm_lahf_lm_dtherm_tpr_shadow_vmai_flexpriority
flags       : fpu_vme_de_pse_tsc_msr_pae_mce_cx8_apic_sep_mtrr_pge_mca_cmov_pat_pse36_clflush_dts_acpi_mmx_fxsr_sse_sse2_ss_ht_tm_pbe_syscall_nx_lm_constant_tsc_arch_perfmon_pebs_bts_rep_good_nopl_aperfperf_pni_dtes64_monit
or_ds_cpl_vmx_est_tm2_ssse3_cx16_xtpr_pcdm_lahf_lm_dtherm_tpr_shadow_vmai_flexpriority
flags       : fpu_vme_de_pse_tsc_msr_pae_mce_cx8_apic_sep_mtrr_pge_mca_cmov_pat_pse36_clflush_dts_acpi_mmx_fxsr_sse_sse2_ss_ht_tm_pbe_syscall_nx_lm_constant_tsc_arch_perfmon_pebs_bts_rep_good_nopl_aperfperf_pni_dtes64_monit
or_ds_cpl_vmx_est_tm2_ssse3_cx16_xtpr_pcdm_lahf_lm_dtherm_tpr_shadow_vmai_flexpriority
```

3 Instalación

El primer paso será actualizar las máquinas:

```
#apt-get update
#apt-get upgrade
#apt-get dist-upgrade
```

Instalación de los paquetes necesarios:

```
#apt-get install qemu-kvm libvirt-bin virtinst kvm bridge-utils
```

Con esto tendremos la primer parte lista para virtualizar, binarios y configuración:

4 Images

Las ISO se añaden al directorio **/var/lib/libvirt/images**. Para actualizar y que la ISO sea visible desde un virt-manager remoto:

```
#virsh
```

Bienvenido a virsh, la terminal de virtualización interactiva.

Ingrese: 'help' para obtener ayuda con los comandos
'quit' para salir

```
virsh# pool-refresh default
```

El grupo default ha sido actualizado

```
virsh# vol-list --pool default
```

Nombre	Ruta
--------	------

ubuntu-14.04.5-server-amd64.iso	/var/lib/libvirt/images/ubuntu-14.04.5-server-amd64.iso
---------------------------------	---

```
virsh# quit
```

Autor: Iván Matarrubias Filgueira	Asunto: KVM/LIBVIRT
	Fecha: 6 de febrero de 2017



5 Creación de máquinas virtuales

Para la creación de las máquinas es muy importante la coherencia, por lo que se han establecido una serie de normas:

- El ID de la máquina será un número de 3 dígitos siendo el primero 101 y así sucesivamente
- La IP de la máquina virtual terminará con el ID de ésta, es decir, si el ID es 104 la IP será 192.168.91.104
- El puerto VNC será $55000 + \text{\$ID_Máquina}$

Script de creación de la máquina de gestión

```
#cat vm_sgestion_crear.sh
```

```
#!/bin/bash
```

```
NOMBRE=sgestion
```

```
DESCR="Gestion de maquinas virtuales"
```

```
ID=199
```

```
CLAVE=sat88861
```

```
TAM_DISCO=8G
```

```
ISO=ubuntu-16.04.1-desktop-amd64.iso
```

```
NET=net-vmbr2
```

```
MACADDR="52:54:00:$(dd if=/dev/urandom bs=512 count=1 2>/dev/null | md5sum | sed 's/^(.)\(.\) \(.\) \(.\) .*$/\1:\2:\3/');
```

```
if [ ! -f /var/lib/libvirt/images/\$ID/\$NOMBRE.qcow2 ]; then
```

```
  qemu-img create -f qcow2 /var/lib/libvirt/images/\$ID/\$NOMBRE.qcow2 \$TAM_DISCO
fi
```

```
# format=iso
```

```
# --disk device=cdrom,path=/var/lib/libvirt/iso/\$ISO,sparse=false,bus=virtio \
```

```
virt-install --hvm --accelerate --noautoconsole \
```

```
--os-type linux \
```

```
--os-variant ubuntu
```

```
--name \$ID-\$NOMBRE \
```

```
--description "\$DESCR" \
```

```
--virt-type kvm \
```

```
--ram 512 \
```

```
--vcpus 1,sockets=1,cores=1 \
```

```
--disk device=disk,format=qcow2,path=/var/lib/libvirt/images/\$ID/
```

```
\$NOMBRE.qcow2,sparse=false,bus=virtio \
```

```
--network network=\$NET,model=virtio,mac=\$MACADDR \
```

```
--graphics vnc,keymap=es,listen=0.0.0.0,port=\$((55000 + \$ID)),password=\$CLAVE \
```

```
--cdrom /var/lib/libvirt/images/\$ISO \
```

```
--boot hd,cdrom
```

```
#--boot hd
```

Autor: Iván Matarrubias Filgueira	Asunto: KVM/LIBVIRT
	Fecha: 6 de febrero de 2017



Script demás máquinas

```
#cat vm_crear.sh
```

```
#!/bin/bash
```

```
NOMBRE=SRV-MSQLTRRV01
```

```
DESCR="Servidor MySQL de tiempo real"
```

```
ID=104
```

```
CLAVE=sat88861
```

```
TAM_DISCO=50G
```

```
ISO=ubuntu-14.04.5-server-amd64.iso
```

```
NET=net-vmbr2
```

```
MACADDR="52:54:00:$(dd if=/dev/urandom bs=512 count=1 2>/dev/null | md5sum | sed 's/^(..)\(..\)(..\).*$/\1:\2:\3/');
```

```
if [ ! -f /var/lib/libvirt/images/$ID/$NOMBRE.qcow2 ]; then
```

```
  qemu-img create -f qcow2 /var/lib/libvirt/images/$ID/$NOMBRE.qcow2 $TAM_DISCO
```

```
fi
```

```
# format=iso
```

```
# --disk device=cdrom,path=/var/lib/libvirt/iso/$ISO,sparse=false,bus=virtio \
```

```
virt-install --hvm --accelerate --noautoconsole \
```

```
--os-type linux \
```

```
--os-variant ubuntuSAUCY \
```

```
--name $ID-$NOMBRE \
```

```
--description "$DESCR" \
```

```
--virt-type kvm \
```

```
--ram 1024 \
```

```
--vcpus 1,sockets=1,cores=1 \
```

```
--disk device=disk,format=qcow2,path=/var/lib/libvirt/images/$ID/
```

```
$NOMBRE.qcow2,sparse=false,bus=virtio \
```

```
--network network=$NET,model=virtio,mac=$MACADDR \
```

```
--graphics vnc,keymap=es,listen=0.0.0.0,port=$((55000 + $ID)),password=$CLAVE \
```

```
--cdrom /var/lib/libvirt/images/$ISO \
```

```
--boot hd,cdrom
```

```
#--boot hd
```

Autor: Iván Matarrubias Filgueira	Asunto: KVM/LIBVIRT
	Fecha: 6 de febrero de 2017



6 Configuración de red

La red por defecto usada por las maquinas virtuales está en la ruta **/etc/libvirt/qemu/networks/default.xml**

```
# cat /etc/libvirt/qemu/networks/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>
```

Sin embargo emplearemos otra, a la que llamaremos **net-vmbr2.xml**

```
# cat /etc/libvirt/qemu/networks/net-vmbr2.xml
```

```
<network>
  <name>net-vmbr2</name>
  <forward mode='bridge'/>
  <bridge name='vmbr2'/>
</network>
```

Implica que tendremos que añadir un nuevo interfaz al servidor de máquinas virtuales que estamos instalando.

```
#nano /etc/network/interfaces
```

```
auto vmbr2
iface vmbr2 inet static
    address 192.168.91.1
    netmask 255.255.255.0
    network 192.168.91.0
    broadcast 192.168.91.255
    bridge_ports none
    bridge_stp off
    bridge_fd 0
```

Tras ello levantaremos el interfaz creado

```
# ifup vmbr2
```

Autor: Iván Matarrubias Filgueira	Asunto: KVM/LIBVIRT
	Fecha: 6 de febrero de 2017



Y finalmente cargaremos el fichero xml de la red y eliminamos el fichero por defecto

```
# virsh net-define net-vmbr2.xml  
# virsh net-start net-vmbr2  
# virsh net-autostart net-vmbr2
```

```
# virsh net-destroy default  
# virsh net-undefine default
```

7 Máquina de gestión

Se trata de una máquina con un ubuntu desktop para, como su nombre indica, gestionar las máquinas virtuales de nuestro entorno.

En esta máquina hay que instalar la herramienta **virt-manager** para la gestión.

```
#apt-get install virt-manager
```

Autor: Iván Matarrubias Filgueira	Asunto: KVM/LIBVIRT
	Fecha: 6 de febrero de 2017



8 Administración de máquinas virtuales qemu/KVM

Virsh es la principal interfaz para la administración de máquinas virtuales qemu/KVM. Virsh es utilizado fundamentalmente para iniciar, pausar y detener máquinas virtuales. También se puede utilizar para obtener un listado de máquinas, su estado e información adicional.

Para operar con virsh podemos abrir su consola de la siguiente forma:

#virsh

También podemos ejecutar de forma directa los comandos sin necesidad de acceder a su consola si lo especificamos al invocar virsh:

#virsh <Comando> <Dominio-id> [OPCIONES]

Tabla 1. Comandos más utilizados en hipervisores Qemu/KVM

Tarea	Comando
Muestra todas las máquinas virtuales disponibles	# virsh list
Iniciar una máquina virtual	# virsh start dominio_id
	a) Apagado normal:
	1. virsh shutdown dominio_id
Detener una máquina virtual	b) Apagado abrupto (simulación de desconexión del cable eléctrico):
	1. virsh destroy dominio_id
Reiniciar una máquina virtual	# virsh reboot dominio_id
Guardar el estado actual de una máquina virtual	# virsh save dominio_id dominio_id-20100102_1830.state
Restaurar una máquina virtual desde un estado previo	# virsh restore dominio_id-20100102_1830.state
Suspender una máquina virtual	# virsh suspend dominio_id
Despertar una máquina virtual suspendida	# virsh resume dominio_id
Montar un dispositivo real en la máquina virtual	# virsh attach-disk dominio_id /u/isos/CentOS-5.4-i386-LiveCD.iso /media/cdrom
Desmontar un dispositivo real de la máquina virtual	# virsh detach-disk dominio_id hdc
Configurar el inicio de una máquina virtual al inicio del servidor	virsh autostart dominio_id
	a) De manera local:
	1. virt-viewer dominio_id
Conectarse a una máquina virtual en ejecución	b) De manera remota:
	1. virt-viewer -c qemu+ssh://direccion_ip_servidor/system dominio_id
Eliminar la definición de una máquina virtual	# virsh undefine dominio_id