Máquinas virtuales en Gentoo

Esta guía describe cómo crear máquinas virtuales en Gentoo. Se utiliza qemu para emular una máquina y, opcionalmente, libvirt, como gestor de máquinas virtuales.

Antes de comenzar, un poco de terminología. Se denomina anfitrión (*host*) al computador en el que se ejecutan una o varias máquinas virtuales. Se denomina huésped (*guest*) a cada uno de los computadores virtuales que se ejecutan en un anfitrión.

Para que un computador pueda hacer de anfitrión de forma eficiente, es necesario asegurarse de que la BIOS, el kernel y el resto del sistema operativo estén configurados correctamente para ello. Dicha configuración se puede consultar, por ejemplo, en la wiki QEMU de Gentoo .

Creación de un disco virtual

Para poner en marcha una máquina virtual, el primer caso consiste en crear un disco virtual. Se puede crear un disco virtual utilizando el siguiente comando:

```
qemu-img create -f qcow2 linuxvm_vda.qcow2 50G
```

Tras ejecutar el anterior comando se habrá creado el fichero `linuxvm_vda.qcow2`. Conviene tener en cuenta que dicho fichero no ocupará inicialmente el tamaño indicado (50 GBytes en el ejemplo anterior), pero podrá crecer hasta dicho tamaño en caso necesario.

Instalación de un sistema operativo

Una vez se dispone de un disco duro virtual, el siguiente paso consiste en instalar sobre dicho disco duro un sistema operativo. La instalación se puede hacer partiendo de una imagen `iso` de un DVD de instalación.

La siguiente línea de comandos lanza una máquina virtual con las siguientes características:

- Memoria: 1GiB.
- Disco duro virtual: `linuxvm_vda.qcow2`.
- ISO de instalación: `ubuntu-14.04.2-desktop-amd64.iso`.
- Arranque desde ISO.
- Acceso a la red a través del anfitrión.

```
/usr/bin/qemu-system-x86_64 \
  -enable-kvm -cpu host \
  -m 1024 \
  -drive file=linuxvm_vda,if=virtio \
  -cdrom ubuntu-14.04.2-desktop-amd64.iso \
  -boot d \
  -net nic,model=virtio -net user \
```

```
  -name "linuxvm"
```

- QEMU/Linux guest

Redirección de puertos USB (flag ``usbredir``, ``spice``?)

Snapshot mode

Configuración de la red (adaptando de kvm a qemu):

- [HOWTO] Setting up KVM with bridged networking

http://wiki.qemu.org/Features-Done/HelperNetworking

```
chown 0640 /etc/qemu/bridge.conf
chown root:kvm /etc/qemu/bridge.conf
```

Descomentar la línea allow br0

gpasswd -a usuario kvm

Disco en /dev/vda

How to resize a qcow2 harddrive image

QEMU/Windows guest

Lanzar con qemu

Una vez creado un fichero de imagen `/home/qemu/lorcb.img`, es posible lanzar una instancia de la máquina virtual con:

```
/usr/bin/qemu-system-x86_64 \
  -enable-kvm -cpu host \
  -drive file=/home/qemu/lorcb.img,if=virtio \
  -net bridge,br=br0 \
  -net nic,model=virtio,macaddr=xx:yy:zz:xx:yy:zz \
  -m 1024 \
  -name "Lorcb" \
  -monitor telnet:127.0.0.1:1111,server,nowait \
  -nographic
```

La opción `-nographic` evita lanzar la interfaz gráfica. Para la instalación del operativo se puede quitar dicha opción y añadir la siguiente `-boot d -cdrom install-amd64-minimal-20141030.iso`.

Instalar libvirt

Para instalar libvirt basta con ejecutar `emerge -av libvirt`. Durante el proceso de instalación se informará de si alguna opción del kernel no está correctamente configurada. En caso de que fuera así, simplemente modificar la configuración del kernel

Convertir un arranque en qemu a un dominio de libvirt

Para conseguir un fichero de definición de una máquina virtual utilizando los parámetros utilizados previamente, se puede utilizar el comando `virsh domxml-from-native qemu-argv`. Por ejemplo, si el comando `qemu-system-x86_64` mostrado anteriormente se copia en el fichero `Lorcb.args`, se podría generar un fichero `Lorcb.xml` adecuado para `libvirt` con el comando:

```
virsh domxml-from-native qemu-argv Lorcb.args > Lorcb.xml
```

Lamentablemente, la conversión puede que no acabe de ir del todo bien. En el ejemplo fue necesario realizar los siguientes cambios en el fichero xml:

1. En el apartado correspondiente al disco, hubo que cambiar su tipo de `raw` a `qcow2`:

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

2. Por otro lado, la red se había configurado para ser del tipo bridge, por lo que hubo que modificar la configuración de red propuesta por la siguiente:

```
 <interface type='bridge'>
    <mac address='xx:yy:zz:xx:yy:zz'/>
    <source bridge='br0'/>
    <target dev='tap0'/>
    <model type='virtio'/>
  </interface>
```

Una vez creado y modificado el fichero `Lorcb.xml`, para definir el dominio `Lorcb`, se debe utilizar el comando:

```
virsh define Lorcb.xml
```

Una vez definido el dominio `Lorcb`, se puede cambiar la definición de la máquina virtual `Lorcb` con el comando:

```
virsh edit Lorcb
```

Arrancar y parar la máquina virtual

Los siguientes comandos arrancan, detienen, resetean y paran una máquina virtual:

```
virsh start Lorcb
virsh shutdown Lorcb
virsh reboot Lorcb
```

```
virsh destroy Lorcb
```

Para hacer que la máquina virtual autoarranque conforme se inicie el demonio `libvirtd`, se puede utilizar el siguiente comando:

```
virsh autostart Lorcb
```

Intefaz gráfica con VNC

Es posible conectarse a la máquina virtual utilizando el comando `vncviewer localhost`, pero para ello se debe haber proporcionado dicha opción en la definición de la máquina virtual (`virsh edit Lorcb`):

```
<graphics type='vnc' port='-1' autoport='yes' listen='127.0.0.1'
keymap='es'>
  <listen type='address' address='127.0.0.1'/>
</graphics>
```

From:
https://lorca.act.uji.es/dokuwiki/ - **Wiki de Lorca**

Permanent link:
https://lorca.act.uji.es/dokuwiki/doku.php/gentoo:qemu?rev=1427202491

Last update: **2015/03/24 13:08**