Administración y virtualización con Linux

Moisés Rubiño García

Documento Personal

ermoi@ermoi.es

11 de noviembre de 2019

Indice

Virtualización en Linux

Virtualización en Linux - ¿Por que?

Un recurso físico, varios lógicos

Es la combinación de hardware y software que permite a una única máquina comportarse como si fueran varias máquinas

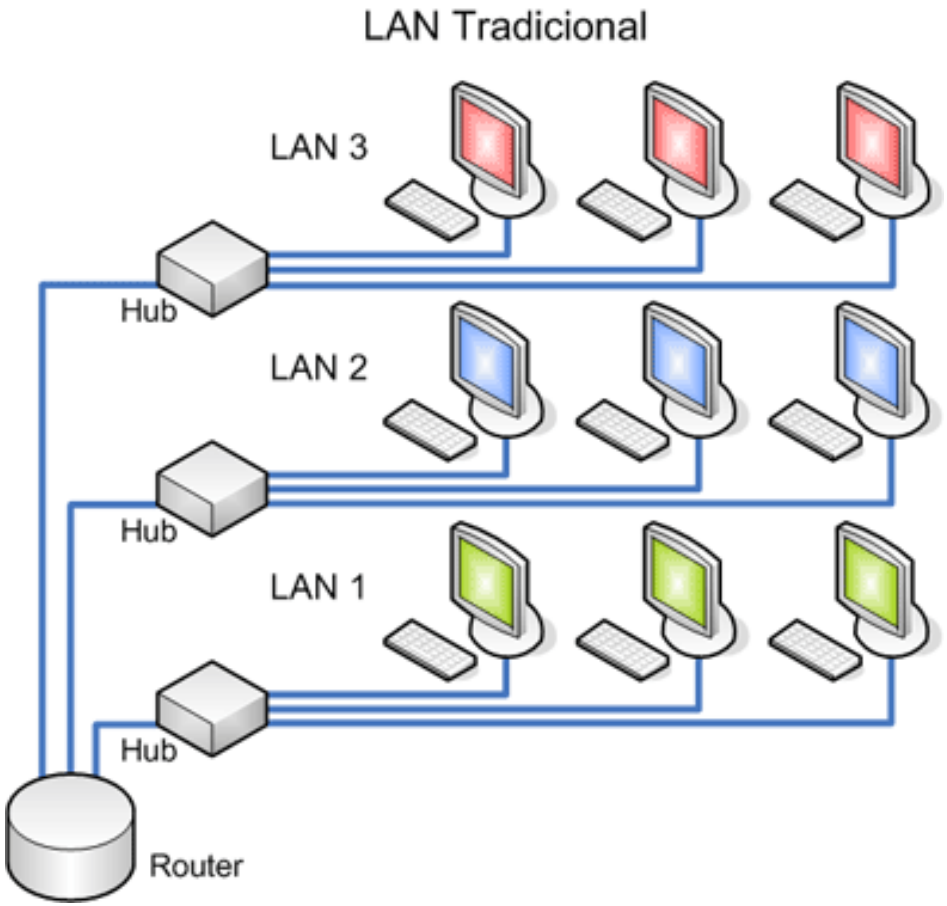
Virtualización en Linux - Características

- El término nace en los 60's y en los 70's IBM ya crea una VM
- Desde 2005 Inter y AMD dan soporte HW para la virtualización, mayor rendimiento
- Permite aislar aplicaciones y usuarios en la mísma máquina
- Capacidad para ejecutar diferentes S.O.
- Permite reducir los costes totales de propiedad
- Minimiza el consumo de energía e infraestructura

Virtualización en Linux - Redes Virtuales

Una red de área local (LAN) esta definida como una red de computadoras dentro de un área geográficamente acotada como puede ser una empresa o una corporación. Uno de los problemas que nos encontramos es el de no poder tener una confidencialidad entre usuarios de la LAN como pueden ser los directivos de la misma, también estando todas las estaciones de trabajo en un mismo dominio de colisión el ancho de banda de la misma no era aprovechado correctamente. La solución a este problema era la división de la LAN en segmentos físicos los cuales fueran independientes entre si, dando como desventaja la imposibilidad de comunicación entre las LANs para algunos de los usuarios de la misma.

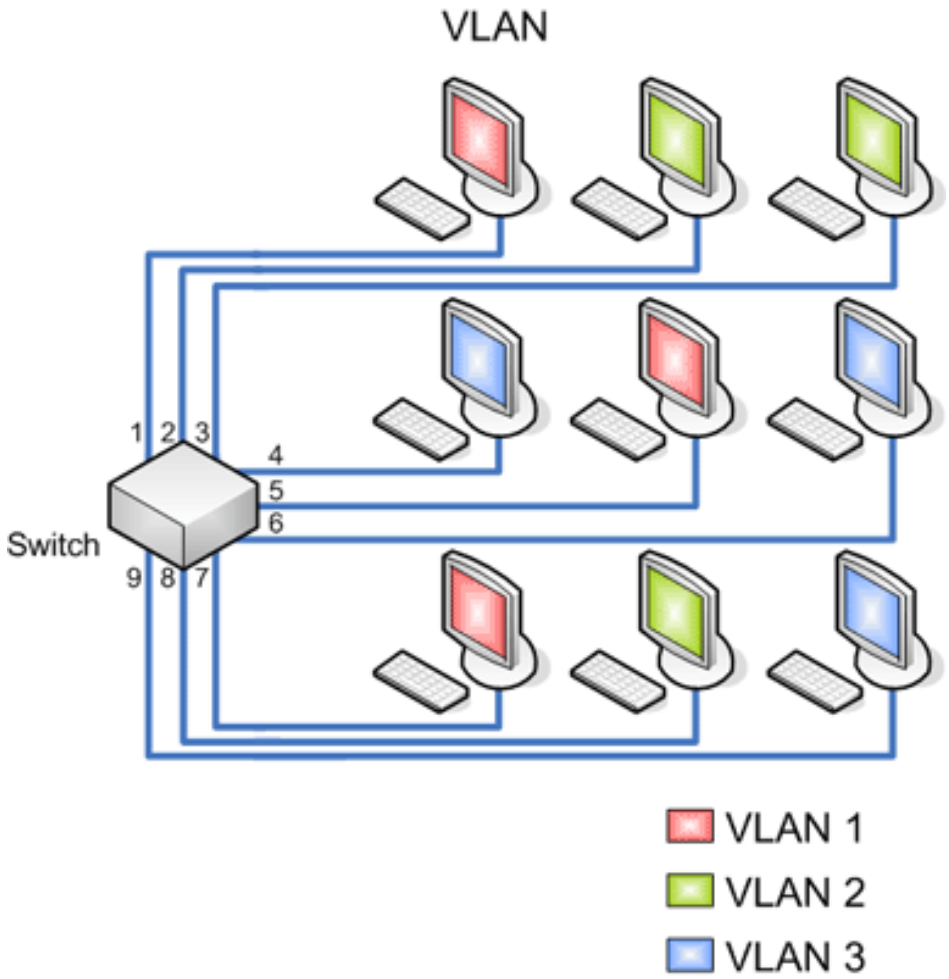
Redes Virtuales - Conceptos



La necesidad de confidencialidad como así el mejor aprovechamiento del ancho de banda disponible dentro de la corporación ha llevado a la creación y crecimiento de las VLANs.
Una VLAN se encuentra conformada por un conjunto de dispositivos de red interconectados (hubs, bridges, switches o estaciones de trabajo). La definimos como como una subred definida por software y es considerada como un dominio de Broadcast que pueden estar en el mismo medio físico o bien puede estar sus integrantes ubicados en distintos sectores de la corporación.
La tecnología de las VLANs se basa en el empleo de Switches, ya que este dispositivo trabaja a nivel de la capa 2 del modelo OSI y es capaz de aislar el tráfico, para que de esta manera la eficiencia de la red entera se incremente.

Una de las ventajas que se pueden notar en las VLAN es la reducción en el trafico de la red ya que solo se transmiten los paquetes a los dispositivos que estén incluidos dentro del dominio de cada VLAN, una mejor utilización del ancho de banda y confidencialidad respecto a personas ajenas a la VLAN, alta performance, reducción de latencia, facilidad para armar grupos de trabajo.
La comunicación que se hace entre switches para interconectar VLANs utiliza un proceso llamado Trunking. El protocolo VLAN Trunk Protocol (VTP) es el que se utiliza para esta conexión, el VTP puede ser utilizado en todas las líneas de conexión incluyendo ISL, IEEE 810.10. IEEE 810.1Q y ATM LANE.

Redes Virtuales - Conceptos



Por puerto

Se configura por una cantidad “n” de puertos en el cual podemos indicar que puertos pertenecen a cada VLAN.

- Ventajas:
  - Facilidad de movimientos y cambios
  - Microsegmentación y reducción del dominio de Broadcast
  - Multiprotocolo: La definición de la VLAN es independiente del o los protocolos utilizados, no existen limitaciones en cuanto a los protocolos utilizados, incluso permitiendo el uso de protocolos dinámicos.
- Desventajas:
  - Administración: Un movimiento en las estaciones de trabajo hace necesaria la reconfiguración del puerto del switch al que esta conectado el usuario.

Por dirección MAC

Los miembros de la VLAN están especificados en una tabla por su dirección MAC

- Ventajas:
  - Facilidad de movimientos: No es necesario en caso de que una terminal de trabajo cambie de lugar la reconfiguración del switch
  - Multiprotocolo
  - Se pueden tener miembros en múltiples VLANs.
- Desventajas:
  - Problemas de rendimiento y control de Broadcast: el tráfico de paquetes de tipo Multicast y Broadcast se propagan por todas las VLAN
  - Complejidad en la administración: En un principio todos los usuarios se deben configurar de forma manual las direcciones MAC de cada una de las estaciones de trabajo

Por protocolo

Asigna a un protocolo una VLAN. El switch se encarga de dependiendo el protocolo por el cual venga la trama derivarlo a la VLAN correspondiente.

- Ventajas:
  - Segmentación por protocolo
  - Asignación dinámica
  - Multiprotocolo: La definición de la VLAN es independiente del o los protocolos utilizados, no existen limitaciones en cuanto a los protocolos utilizados, incluso permitiendo el uso de protocolos dinámicos.
- Desventajas:
  - Problemas de rendimiento y control de Broadcast: Por las búsquedas en tablas de pertenencia se pierde rendimiento en la VLAN
  - No soporta protocolos de nivel 2 ni dinámicos

Por direcciones IP

Esta basado en el encabezado de la capa 3 del modelo OSI. Las direcciones IP a los servidores de VLAN configurados. No actúa como router sino para hacer un mapeo de que direcciones IP están autorizadas a entrar en la red VLAN. No realiza otros procesos con la dirección IP.

- Ventajas:
  - Facilidad en los cambios de estaciones de trabajo: Cada estación de trabajo al tener asignada una dirección IP en forma estática no es necesario reconfigurar el switch.
- Desventajas:
  - El tamaño de los paquetes enviados es menor que en el caso de utilizar direcciones MAC.
  - Perdida de tiempo en la lectura de las tablas
  - Complejidad en la administración: En un principio todos los usuarios se deben configurar de forma manual las direcciones MAC de cada una de las estaciones de trabajo.

Por nombre de usuario

Se basan en la autenticación del usuario y no por las direcciones MAC de los dispositivos.

- Ventajas:
  - Facilidad de movimiento de los integrantes de la VLAN
  - Multiprotocolo
- Desventajas:
  - En corporaciones muy dinámicas la administración de las tablas de usuarios

Acceso a switch

- Por consola:
  - a través de minicom (linux) o hyperterminal (windows)
  - El puerto serie es un bien muy preciado
  - Tipos de cables de consola
- Por telnet
- Por ssh
- Por web

Se recomienda siempre una vlan aislada y sin acceso a la calle para el control de la electrónica de red

El estandar cisco

Comandos básicos

- show running-config
- show startup-config
- copy running-config startup-config
- show vlan
- configure t

Ejemplos

- configurar puertos en diferentes vlan
- configurar puertos en trunk
- configurar puertos con vlan nativa (untagged) y tagged

En general no conviene estar ejecucanto NetworkManager en server. Antes de nada, "systemctl stop NetworkManager.service" y "systemctl disable NetworkManager.service"

Activación soporte vlan

- Fichero /etc/sysconfig/network
- meter la línea VLAN=yes

Configuración de los interfaces

- Ficheros /etc/sysconfig/network-scripts/ifcfg-nombreinterfaz.numerovlan
- dentro del fichero DEVICE=nombreinterfaz.numerovlan
- asegurarse que aparece NM_CONTROLLED="no" (es redundante con parar el servicio de Network Manager, pero por si acaso)

Principales tipos de interfaces de red

- normal,bonding, bridge, virtuales

Bridge

Configuración. Hay que poner en la tarjeta de red madre (eth0) la entrada "BRIDGE=br0"

Y a continuación, crear una nueva tarjeta de red ifcfg-br0 con el siguiente contenido

```
DEVICE=br0
TYPE=Bridge
BOOTPROTO=none
NM_CONTROLLED="no"
ONBOOT=yes
```

Configurando bonding

Poner en **los** interfaces madre (eth0, eth1) las siguientes líneas, "MASTER=bond0" y "SLAVE=yes"
Luego hay que crear un fichero ifcfg-bond0 con el contenido

```
DEVICE=bond0
BOOTPROTO=none
#BONDING_OPTS="mode 4 miimon=100"
BONDING_OPTS="mode=802.3ad miimon=100"
NM_CONTROLLED="no"
#ONBOOT=yes
STARTMODE="onboot"
BONDING_MASTER="yes"
BONDING_SLAVE0="eth0"
BONDING_SLAVE1="eth1"
```

Tipos de bonding

Hay cambio de configuración entre Centos 6 y 7. La única diferencia, es cuando definimos el tipo de bonding, en Centos 6 BONDING_OPTS="mode 4 miimon=100" y Centos 7 BONDING_OPTS="mode=802.3ad miimon=100". Los tipos son:

- mode=**balance-rr** or **0** (Balance Round Robin)
- mode=**active-backup** or **1** (Active backup) ←
- mode=**balance-xor** or **2** (Balance XOR)
- mode=**broadcast** or **3** (Broadcast)
- mode=**802.3ad** or **4** (802.3ad) ←
- mode=**balance-tlb** or **5** (Balance TLB)
- mode=**balance-alb** or **6** (Balance ALB)

En el bonding activo-activo hay que activar LACP en los puertos del bonding del switch

Consideraciones

- En caso de querer montar bonding, vlan y bridge, el orden es bond de todas las interfaces, luego separamos las interfaces y sobre cada interfaz de vlan, hacemos los bridge.
- Las tarjetas bridge o bonding puede o no tener ip.
- Verificación de funcionamiento de bonding: /proc/net/bonding/bond0
- NetworkManager (nmcli) no recomendado

Referencias de modos de bonding

- http://systemadmin.es/2009/04/los-modos-de-bonding
- http://www.cloudibee.com/network-bonding-modes/
- https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/7/html/Networking_Guide/sec-Using_Channel_Bonding.html
- http://www.linuxfoundation.org/collaborate/workgroups/networking/bonding

Acciones importantes a realizar en la postinstalación de centos

- **Desactivar SELINUX**, para evitar problemas. Para ello hay que editar el fichero **/etc/selinux/config** y sustituir SELINUX=enforcing por **SELINUX=disabled**
- Desactivar el servicio de NetworkManager, para ello, hay que ejecutar "systemctl stop networkmanager" y "systemctl disable networkmanager"
- Además se recomienda que en todos los ficheros de configuración de los interfaces, aparezca la entrada "NM_CONTROLLED=no"

Redes Virtuales - VLAN en linux - Comandos

- ip
- telnet
- tcpdump -nvl -i
- iftop -i
- traceroute
- tcptraceroute
- dig @
- whois
- nmap (zenmap)

Referencias

- http://linoxide.com/linux-command/use-ip-command-linux/
- http://www.tecmint.com/ip-command-examples/

Virtualización en Linux - Almacenamiento virtual

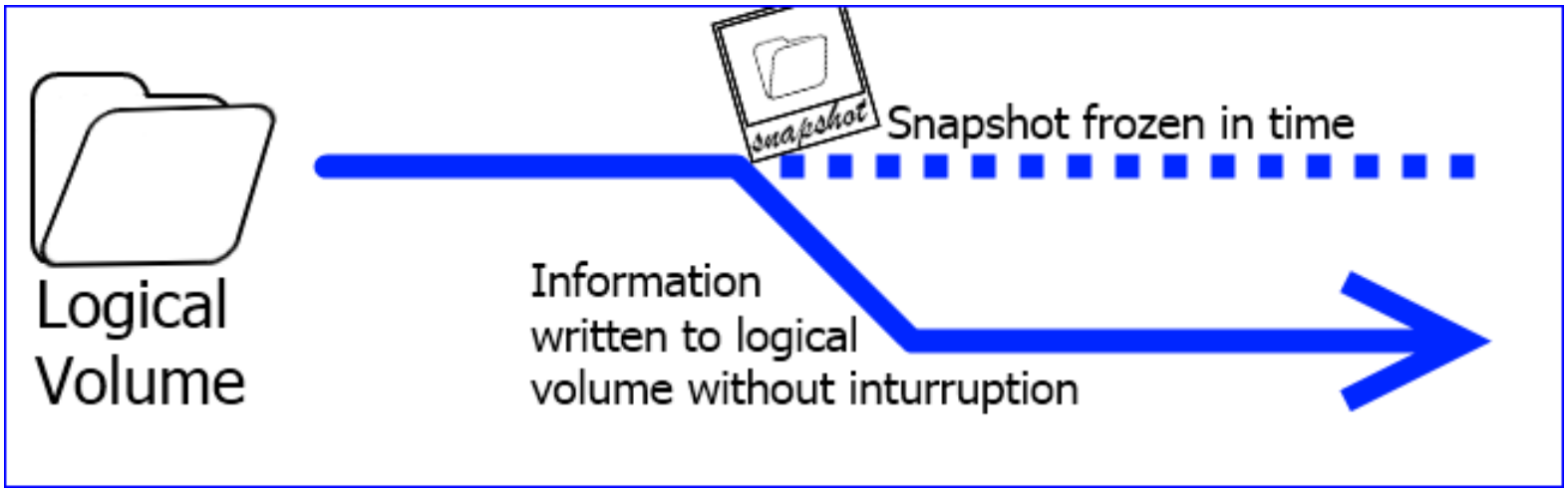
Almacenamiento virtual - Conceptos

Gestión virtual de los discos.

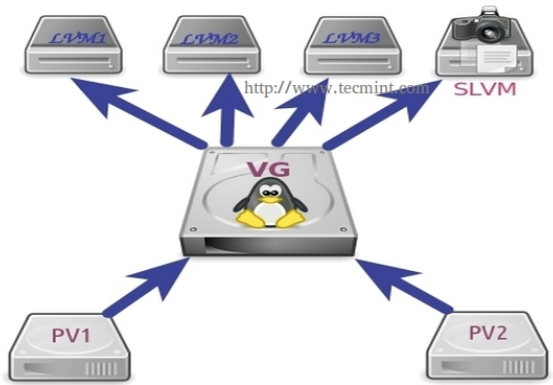
- Se puede hacer raid (con lvcreate –type)
- Se puede decidir que volumen lógico esta en que disco, por si hay de distintas velocidades
- Se pueden crear con la opción thinprovisioning (no recomendado para pro en sistemas de ficheros "cerrados")
- Se pueden realizar snapshot.

El sistema de ficheros Btrfs o ZFS combina LVM y RAID

Almacenamiento virtual - Snapshots



Almacenamiento virtual - Snapshots



Un snapshot en LVM no es más que un grupo de volúmenes lógico y debe de ser tratado como tal. Se puede expandir.
En caso de que se llene, deja de ser funcional y debe de eliminarse cuanto antes. Hay opciones de autoeliminación en caso de llenado.

Almacenamiento virtual - Snapshots

Funcionamiento para hacer backup

- "lvcreate -s -L 100M -n snaplv1 /dev/vg1/lv1" (lvcreate –size 100M –snapshot –name snaplv1 /dev/vg1/lv1) Creamos el snapshot snaplv1 de 100M del volumen lógico lv1. El volumen lógico se crea en el mismo volumen físico del lv1, a saber, vg1, por tanto tenemos que tener espacio para crearlo.
- hemos añadido el disco b al volumen físico para ahora usarlo para los snapshot (lvcreate –size 100M –snapshot –name snaplv1 /dev/vg1/lv1 sdb)
- Comprobamos "lvs"
- Backup del volumen. Para ello lo montamos "mount /dev/vg1/snaplv1 /snapshot" (mount: block device /dev/vg1/snaplv1 is write-protected, mounting read-only)
- Hacemos el backup "tar -cf /dev/rmt0 /puntodemontajeoriginal"
- Y ahora eliminamos el snapshot. Para ello primero desmontamos "umount" y luego eliminamos "lvremove /dev/vg1/snaplv1"

Almacenamiento virtual - Snapshots

Funcionamiento para hacer probar cambios en sistema de ficheros, por ejemplo actualización

- "lvcreate -s -L 100M -n snaplv1 /dev/vg1/lv1" (lvcreate –size 100M –snapshot –name snaplv1 /dev/vg1/lv1) Creamos el snapshot snaplv1 de 100M del volumen lógico lv1. El volumen lógico se crea en el mismo volumen físico del lv1, a saber, vg1, por tanto tenemos que tener espacio para crearlo.
- Para revertir cambios. No montamos el snapshot y ejecutamos "lvconvert –merge /dev/vg1/snapshotlv1"
- En el caso de que el volumen lógico que que estamos echando para atras este montado, nos dará un error avisandonos de que no sera efectivo hasta que reiniciemos. Lo suyo es reiniciar y sino, desmontar antes de aplicar el merge

Almacenamiento virtual - Snapshots

Es importante mencionar que si el snapshot se queda sin espacio, este será inservible y no podrá ser usado. Simplemente monitorizando su tamaño podremos vigilar el estado de este, y en caso de que se llene, se podría ampliar sin problema, como si de un volumen lógico normal y corriente se tratara.

Virtualización en Linux - Kernel Virtualization Mode

Kernel Virtualization Mode - Conceptos

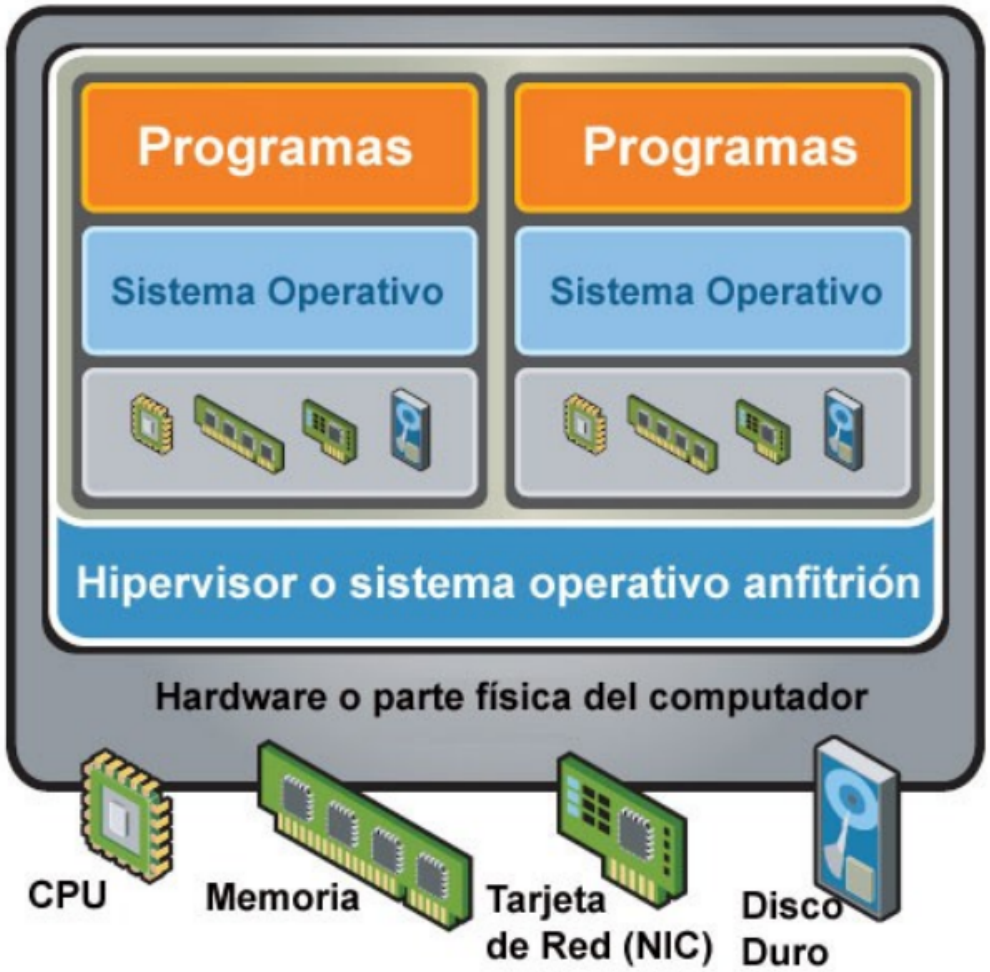
Hypervisores

- Es como se le conoce al software de virtualización
- Son los encargados de gestionar las diferentes VM
- Se encargan de aislar las diferentes VM
- Ofrecen una interfaz única de acceso al hardware

Kernel Virtualization Mode - Terminos

- **P2V**: Phisical to Virtual: Migrar de físico a virtual
- **V2V**: Virtual to Virtual: Migrar de virtual a virtual (de un hipervisor a otro)
- **V2P**: Virtual to Phisical: Migrar de virtual a físico
- **Dominio**: máquina virtual

Kernel Virtualization Mode - Conceptos



Kernel Virtualization Mode - KVM

- Sistema "oficial" de virtualización del kernel (aunque xen ya esta integrado)
- Incluido desde la versión 2.6.20
- Es un módulo del kernel que convierte el SO Linux en un hypervisor
- Utiliza el "API estandar" del kernel
- Necesidad de soporte en el procesador de instrucciones IntelVT, AMDV
- Soporte de drivers paravirtualizados para mejorar el rendimiento

Kernel Virtualization Mode - KVM

Seguridad

- Cada Virtual Machine es un proceso
- Cada CPU es un thread
- Se aprovecha de modelo seguridad selinux/AppArmor
- Svirt

Es sw libre

Kernel Virtualization Mode - Almacenamiento

- raw
  - Ocupa todo el tamaño total
  - Cada CPU es un thread
  - No soporta snapshots, compresión ni cifrado
  - Mejor I/O
- qcow2
  - Imagen ocupa tamaño real y aumenta según necesidad
  - Soporta snapshots, compresión y cifrado

Kernel Virtualization Mode - KVM Drivers

KVM soporta virtualización hibrida

- IDE
  - En los SO invitados: emulación drivers
  - Bajo I/O en dispositivos de bloques y red
- VirtIO
  - En los SO invitados drivers paravirtualizados
  - Alta I/O en dispositivos de bloques y red
  - VirtIO es una interfaz independiente del hypervisor
  - Incluido en kernel ¿ 2.6.25
  - RedHat ha desarrollado drivers virtIO para Windows, certificados por Microsoft.

Kernel Virtualization Mode - KVM Drivers

Como montar virtio drivers para disco duro para win32

1. Montar el disco del SO en IDE
2. Instalar el SO
3. Añadirle un disco de 1G tipo Virtio
4. Instalar los drivers de Virtio para el nuevo disco
5. Eliminar el nuevo disco
6. Cambiar el tipo de disco del SO de IDE a Virtio

Kernel Virtualization Mode - Referencias

KVM

- http://www.linux-kvm.org/page/Main_Page

Virtio

- http://wiki.libvirt.org/page/Virtio
- https://fedorapeople.org/groups/virt/virtio-win/direct-downloads/stable-virtio/

Virtualización en Linux - Linux containers (LXC)

Linux containers (LXC) - Conceptos

Es una tecnología de "virtualización" en el nivel de sistema operativo (SO) para Linux. LXC permite que un servidor físico ejecute múltiples instancias de sistemas operativos aislados, conocidos como Servidores Privados Virtuales (SPV o VPS en inglés) o Entornos Virtuales (EV). LXC no provee de una máquina virtual, más bien provee un entorno virtual que tiene su propio espacio de procesos y redes.
Es similar a otras tecnologías de virtualización en el nivel de SO como OpenVZ y Linux-VServer, asimismo se asemeja a aquellas de otros sistemas operativos como FreeBSD jail y Solaris Containers.

Linux containers (LXC) - ¿En que se basa?

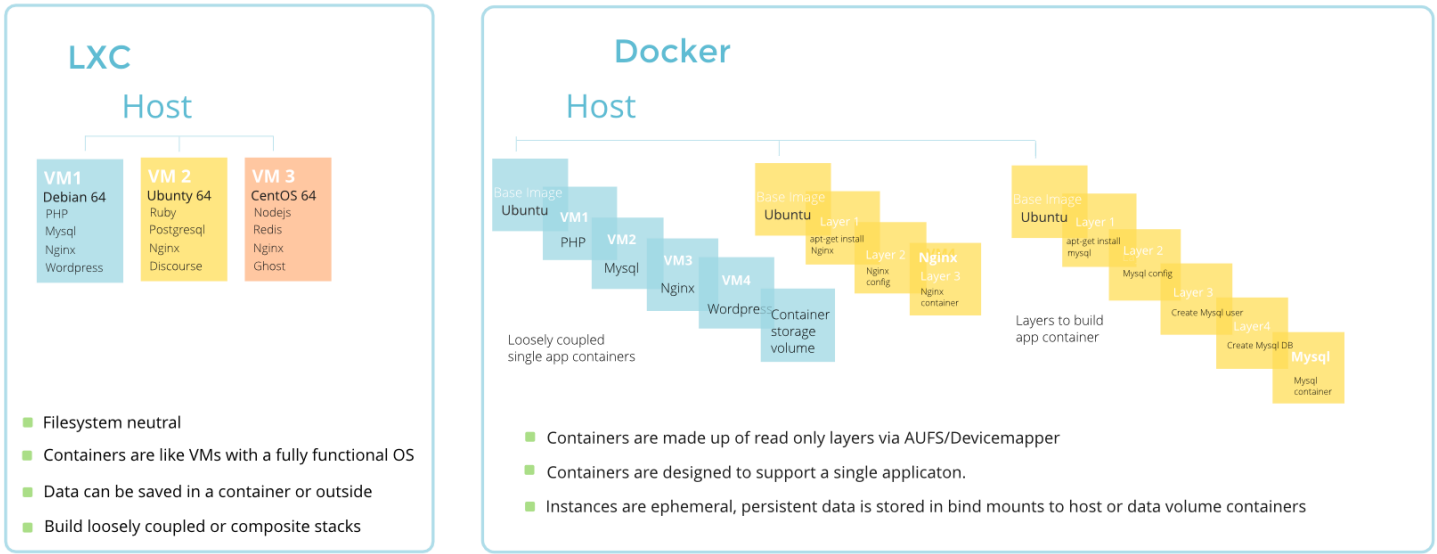
LXC se basa en la funcionalidad cgroups del Linux que está disponible desde la versión 2.6.29, desarrollada como parte de LXC. También se basa en otras funcionalidades de aislamiento de espacio de nombres, que fueron desarrolladas e integradas dentro de la línea principal del núcleo de Linux. A veces se dice que LXC es como chroot con esteroides.

Linux containers (LXC) - Características

La tecnología de LinuX Containers te permite crear unas cosas (contenedores o VPS) que son como servidores linux completamente aislados, en tu máquina Linux, compartiendo el kernel con ella. Es como una virtualización muy ligera, tan ligera que realmente no hay virtualización en absoluto, y por lo tanto no tiene un impacto negativo en el rendimiento.
Se pueden ejecutar diferentes distribuciones en los contenedores, siempre y cuando usen el mismo kernel que el la máquina donde están.

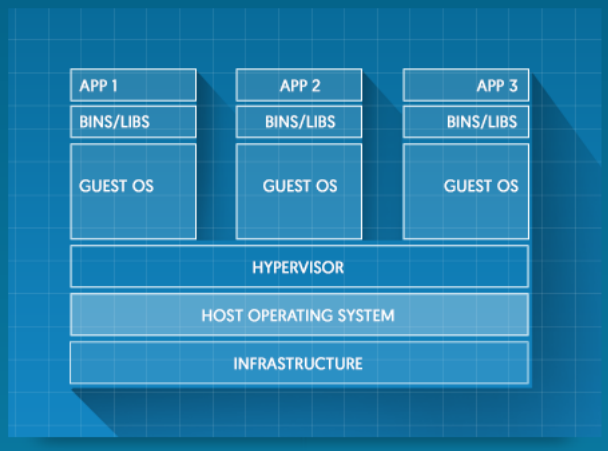
Dockers es una vuelta de tuerca más y solo aisla la parte de sw en capas de forma que un contenedor de dockers solo puede albergar una aplicación. Se usa en situaciones en lo que lo importante es "virtualizar" la aplicación, pero en la que se puede compartir no solo núcleo, sino también sw base para que corra la aplicación. Por ejemplo un contenedor para una app de python y otra para otra app de python, pero ambas corriendo bajo la misma versión de python. En caso de que queramos tener varias versiones de python, deberíamos ir a LXC. Además en dockers prima el espacio, ya que se usa el mismo sw base y cada contenedor solo tiene el sw de la app que corre.

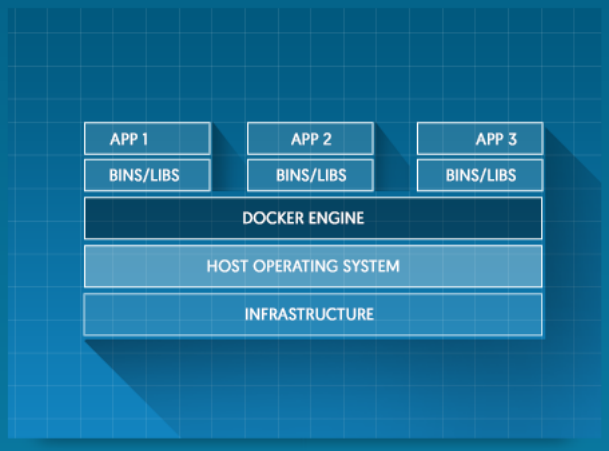
Key differences between LXC and Docker



COMPARING CONTAINERS AND VIRTUAL MACHINES

Containers and virtual machines have similar resource isolation and allocation benefits -- but a different architectural approach allows containers to be more portable and efficient.





VIRTUAL MACHINES

Virtual machines include the application, the necessary binaries and libraries, and an entire guest operating system -- all of which can amount to tens of GBs.

CONTAINERS

Containers include the application and all of its dependencies --but share the kernel with other containers, running as isolated processes in user space on the host operating system. Docker containers are not tied to any specific infrastructure: they run on any computer, on any infrastructure, and in any cloud.

Linux containers (LXC) - Instalación

Simplemente hay que ejecutar

- yum -y install epel-release
- yum install lxc lxc-templates libvirt debootstrap
- systemctl start lxc.service (para levantar los servicios)
- systemctl start libvirtd
- lxc-checkconfig (para chequear que todo esta ok)

Localización de ficheros

- Todas los container se almacenan en "/var/lib/lxc" (por defecto)
- Es posible usar backend de LVM. Al igual que en KVM se le pasa un VG y se van creando por cada container un LV. El fichero donde se especifica el VG es lxc.conf. El tipo de sistema de ficheros y el tamaño se especifica en los parametros que se le pasan a lxc-create.
- Los ficheros de configuración global de lxc se almacenan en /etc/lxc (y en ./.config/lxc/)

Linux containers (LXC) - Networking

- Por defecto LXC usa una red privada para cada container. Normalmente usan esta red privada para salir al exterior a través de NAT a la tarjeta de red.
- Si usando la configuración por defecto (red privada) queremos tener ips fijas, hay que insertar entradas en el fichoero /etc/lxc/dnsmasq.conf
  1. dhcp-host=lxcmail,10.0.3.100
  2. dhcp-host=ttrss,10.0.3.101

- Es posible configurar sin la red privada, conectandose directamente a la interfaz de red. Para ello lo recomendable es usar interfaces de red bridge. Para ello es necesario cambiar la configuración del contenedor poniendo
  - lxc.network.type = veth
  - lxc.network.link = br0
- También se puede hacer directamente, la configuración sería
  - lxc.network.type = phys
  - lxc.network.hwaddr = 00:16:3e:c6:0e:04
  - lxc.network.flags = up
  - lxc.network.link = tap0
  - lxc.network.name = eth0

Linux containers (LXC) - Creación de containers

Verificamos las plantillas de las que disponemos

- ls -alh /usr/share/lxc/templates/

Y para crear un container solo hace falta

- lxc-create -n container_name -t container_template (lxc-create -n nombremaquina -t debian)
- lxc-create −n mywheezy −t debian −− −r wheezy −a amd64
- lxc-create (opciones)
  - −n = name
  - −t = template
  - −d = distibution
  - −a = arch
  - −r = release
- lxc-create −n centos7lvm −t centos −B lvm −−vgname centos −−fssize 5G −−fstype xfs (para crear en LVM)
- lxc-ls o (para obtener listado de contenedores) o lxc-ls −−active (para ver solo los activos)
- lxc-info (muestra información de un contenedor, con su ip ...)

Linux containers (LXC) - Configuración container

Todos los container se almacen en "/var/lib/lxc", creando un directorio para cada container. Dentro de ese directorio, la configuración del container esta en un fichero llamado config. Veamos algunos de esos parámetros

- lxc.network.type (controla que tipo de red tendrá el container. Por defecto es "veth" - virtual enthernet pairs)
- lxc.network.veth.pair (el nombre de la interface veth que se creará en la madre. Si no se pone nada, lo crea de forma aleatoria)
- lxc.network.link (especifica el bridge que se usará para enganchar el veth. Si no se pone nada, nat)
- lxc.rootfs (especifica donde esta el almacenaje del container. Por defecto /var/lib/lxc/¡container name¿/rootfs)

Linux containers (LXC) - Establecer contraseña

Hay dos opciones

- cat /var/lib/lxc/centos_lxc/tmp_root_pass (mostramos la contraseña temporal puesta)
- chroot /var/lib/lxc/centos_lxc/rootfs passwd (establecemos la que queremos)

Linux containers (LXC) - Manejo de containers

Arrancar un container

- lxc-start -n mydeb -d (para arrancar un container)

Conexión a un container

- lxc-console -n mydeb
- Para desasociar una consola, "Ctrl+a" y luego "q"

Para parar un container

- lxc-stop -n mydcb

Para eliminar un container

- lxc-destroy -n mywheezy

Para pausar un container

- lxc-freeze -n container_name (lxc-unfreeze -n container_name)

Para clonar un container

- lxc-clone mydeb mydeb-clone

Se pueden tomar snapshot

- lxc-snapshot

Linux containers (LXC) - Configuración personalizada para Centos

- Si la hija es un Centos 7 (madre Cento 7 también) hay que poner en las hijas, en el fichero de configuración de la máquina hija "lxc.kmsg = 0". Esto evita que haya problemas de gestión del disco. Si no lo pones al hacer update y demás se queda tonta.
- Si la madre en un Centos 7, en la configuración general de la plantilla de centos "/usr/share/lxc/config/centos.common.conf" hay que quitar un parametro para poder instalar el http. No hace falta para el nginx. En vez de poner "lxc.cap.drop = mac_admin mac_override setfcap" hay que poner "lxc.cap.drop = mac_admin mac_override" (eliminar el "setfcap"

Para realizar un migrado de un contenedor, hay que hacer un tar y luego descomprimirlo, pero teniendo cuidado de que mantenga los uids (opción –numeric-owner)

- tar −−numeric-owner −czvf lxc.tgz /lxc/my_lxc (para crear el tar)
- tar −−numeric-owner −xzvf lxc.tgz (para descomprimir el tar)

Linux containers (LXC) - Limitando los recursos

Para establecer los recursos

- lxc.cgroup.cpuset.cpus = 0,3 (asigna el procesador 0 y 3 a la máquina)
- lxc.cgroup.memory.limit_in_bytes = 512M (total, memoría + swap)
- lxc.cgroup.memory.memsw.limit_in_bytes = 1G (total, memoría + swap)

Para medir los recursos

- /sys/fs/cgroup/memory/lxc/maquina/... (por máquina)
- /sys/fs/cgroup/memory/lxc... (global)

Linux containers (LXC) - Autostart

Es posible marcar container para que se arranque al inicio de una máquina. Para ello, en el fichero de configuración de la máquina añadimos

- lxc.start.auto = 1
- lxc.start.delay = 5 (que pasen 5 segundos desde que arranca un container a otro)

Linux containers (LXC) - Referencias

- http://www.tecmint.com/install-create-run-lxc-linux-containers-on-centos/
- http://www.itzgeek.com/how-tos/linux/centos-how-tos/setup-linux-container-with-lxc-on-centos-7-rhel-7.html
- https://help.ubuntu.com/lts/serverguide/lxc.html
- http://blog.scottlowe.org/2013/11/25/a-brief-introduction-to-linux-containers-with-lxc/
- https://www.stgraber.org/2013/12/20/lxc-1-0-blog-post-series/
- https://www.flockport.com/guides/
- https://linuxcontainers.org/
- http://www.jpablo128.com/por-que-usar-lxc-linux-containers/

Virtualización en Linux - Diferencia entre máquinas virtuales y contenedores

Diferencias entre KVM y LXC - Conceptos

The difference between LXC and KVM virtualization is that LXC doesn't emulates hardware, but shares the same kernel namespace, similar to chroot applications.
Simular mediante software TODO el hardware que la máquina virtual utiliza. Emulación
Virtualización. El rendimiento es elevado pero todos los VPS tienen que ser utilizando el mismo kernel.

Diferencias entre KVM y LXC - ¿Como decidir?

Resumen

A la hora de elegir el sistema hay que tener en cuenta que cuanto mayor sea el aislamiento entre/de máquinas virtuales menor será el rendimiento.

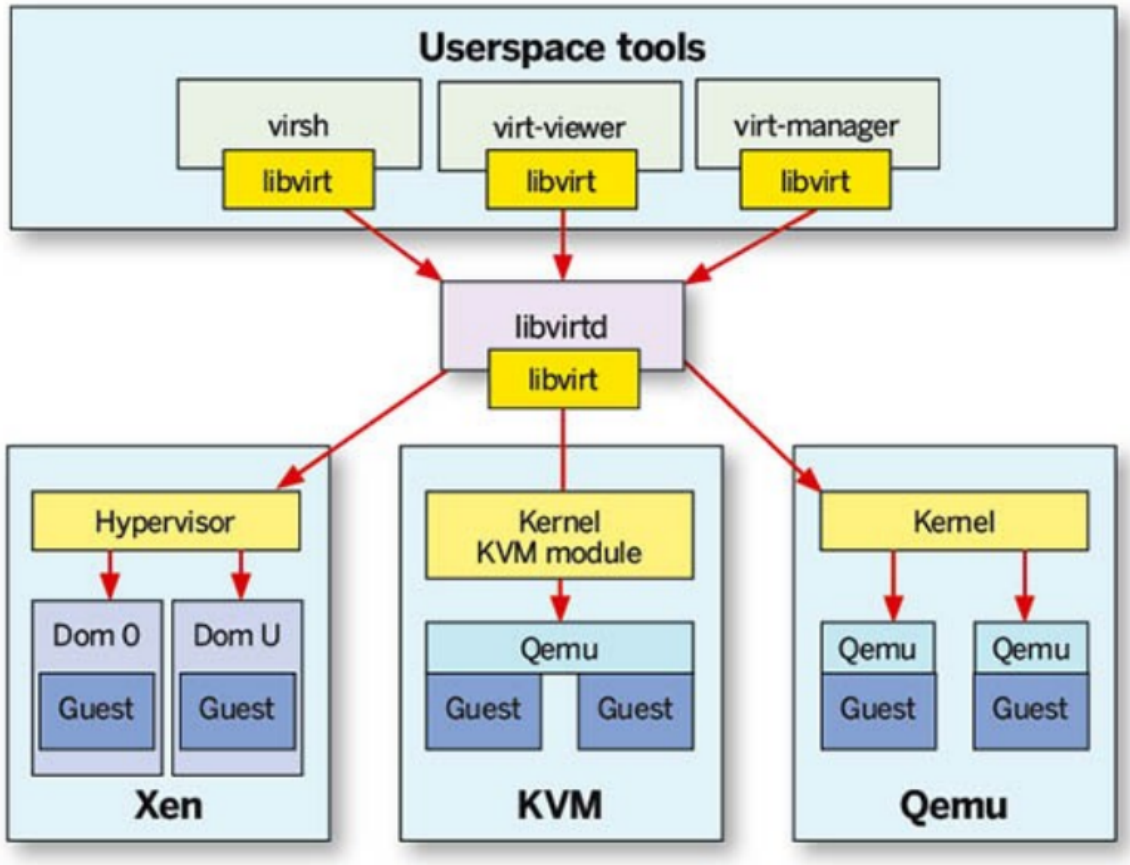
Virtualización en Linux - libvirtd, administración de entornos virtuales

Look into libvirt

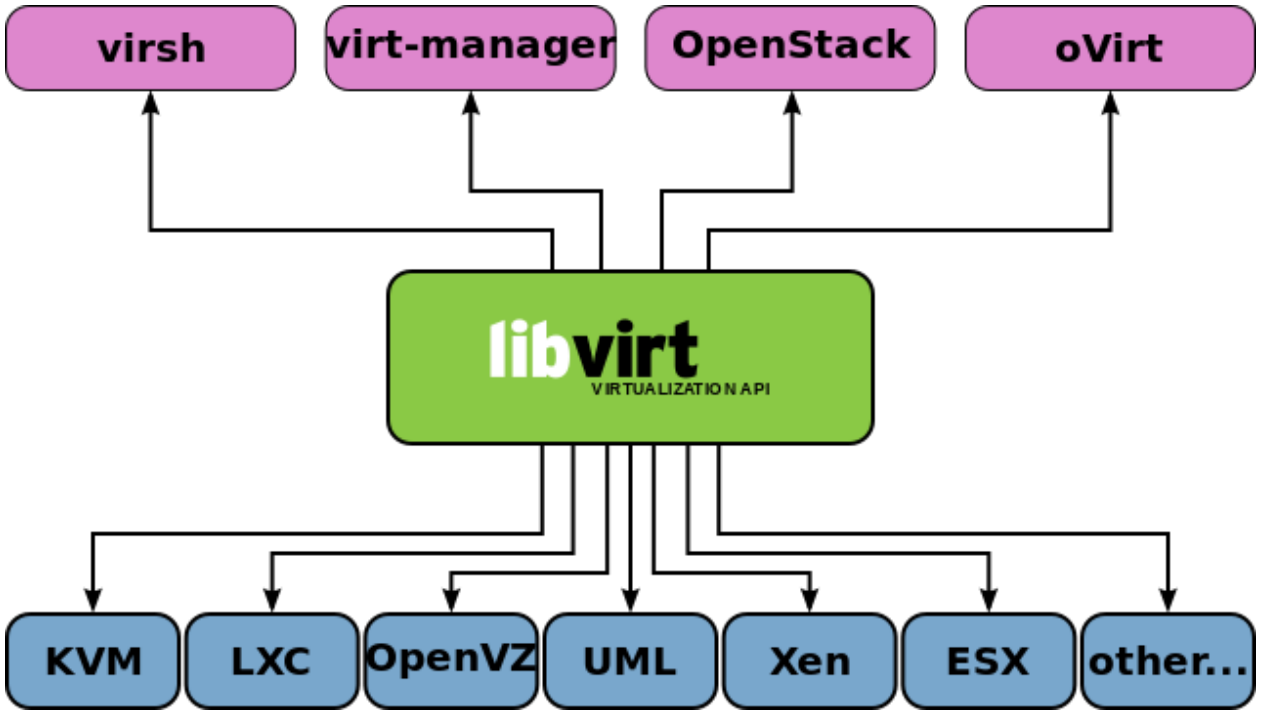
libvirtd - Conceptos

- Conjunto de herramientas con API para virtualización de sistemas Linux
- Soporta los siguientes hypervisores KVM/QEMU, XEN, LXC, OpenVZ, UML, Virtualbox, Vmware ESX/GSX, Workstation, Hyper-V
- Almacenamiento IDE/SCSI/USB/, LVM, iSCSI, NFS..
- Software Libre

libvirtd - Conceptos



libvirtd - Conceptos



libvirtd - ficheros XML

Virsh puede gestionar ficheros XML. Es muy útil para realizar scripting avanzado para grandes despliegues

```
# cat satelliteiso.xml
<disk type="file" device="disk">
        <driver name="file"/>
        <source file="/var/lib/xen/images/rhn-satellite-5.0.1-11-redhat-linux-as-i386-4-embedded-oracle.iso"/
        <target dev="hdc"/>
        <readonly/>
</disk>
```

Run **virsh attach-device** to attach the ISO as **hdc** to a guest called "satellite" :

```
# virsh attach-device satellite satelliteiso.xml
```

libvirtd - PATH

- Los ficheros de configuración de las máquinas virturales suelen estar en /etc/libvirtd/qemu
- Los discos duros
  - En local, usualmente en: /var/lib/libvirt/images, aunque podemos personalizarlo
  - En remoto: nfs, samba, iscsi, fibre channel

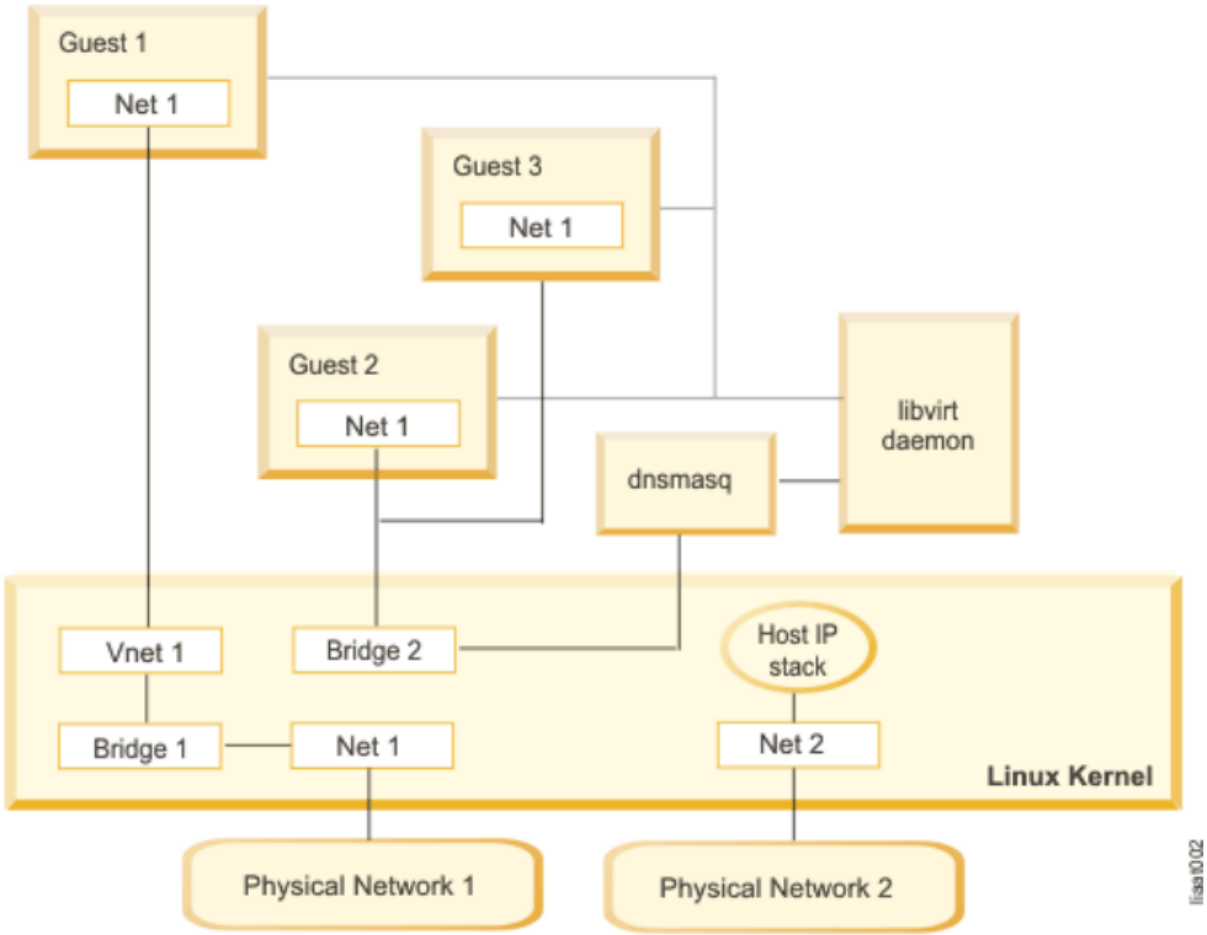
libvirtd - Instalación

- Solo es necesario instalar los siguientes paquetes: qemu-kvm, libvirt, libvirt-python, qemu-img, virt-v2v, libguestfs
- reiniciar el servicio de libvirtd (system restart libvirtd)

libvirtd - Virtual Networking

- Para la implementación de redes virtuales libvirt utiliza el concepto de switches virtuales
- El anfitrión Linux representa un switch virtual como una interfaz de red.
- Default es un switch virtual representado por virbr0
- Tenemos una interfaz virbr0 que es la encargada de la conectividad NAT
- Por defecto con el NAT tienen acceso de red a otras máquinas virtuales o al host a través de 192.168.122.0
- Si se quiere que las máquinas tengan acceso a la LAN es necesario crear un BRIDGE

libvirtd - Virtual Networking

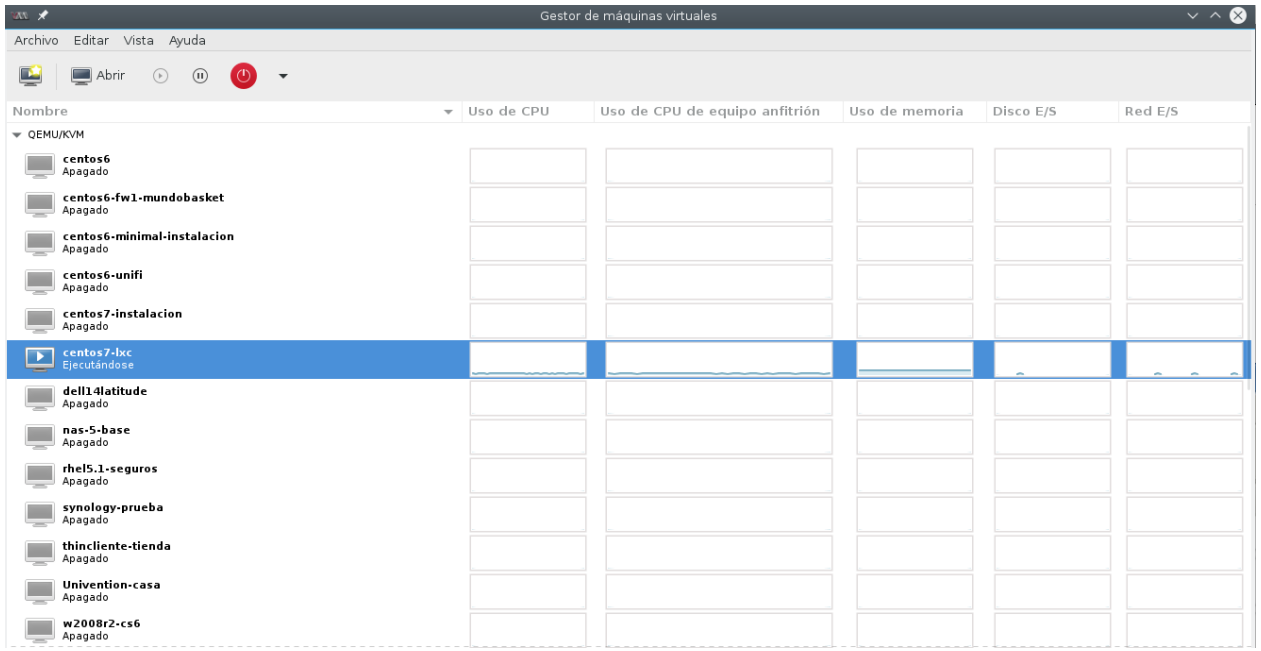


Virtualización en Linux - virt-manager, interfaz gráfico de administración

virt-manager - Conceptos

- Herramienta para gestionar las máquinas virtuales, consultar consumo de recursos, acceder a consolas virtuales, clonación de máquinas
- Permite gestionar hypervisores QEMU, KVM, XEN
- Permite conexiones locales y remotas
- Muestra estadísticas de rendimiento
- Permite gestión de pools de almacenamiento

virt-manager - Conceptos



virt-manager - Creación de VM

- A través del interfaz
- editar el XML
- opciones posibles

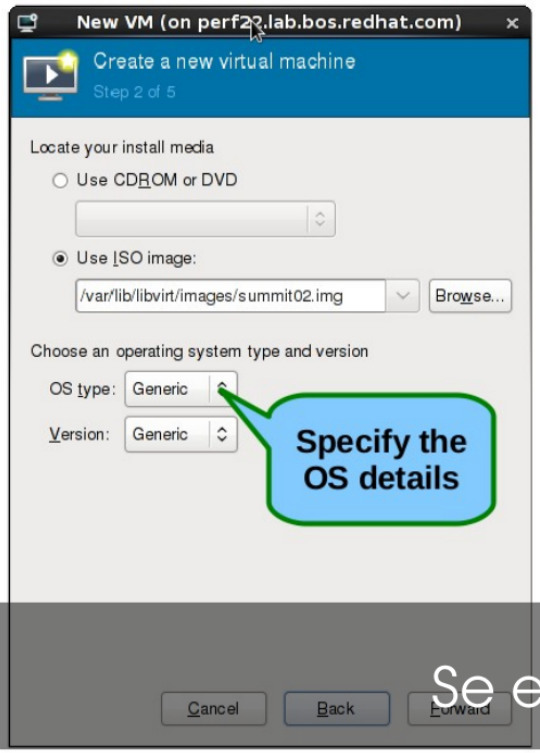
virt-manager - Gestión de pools

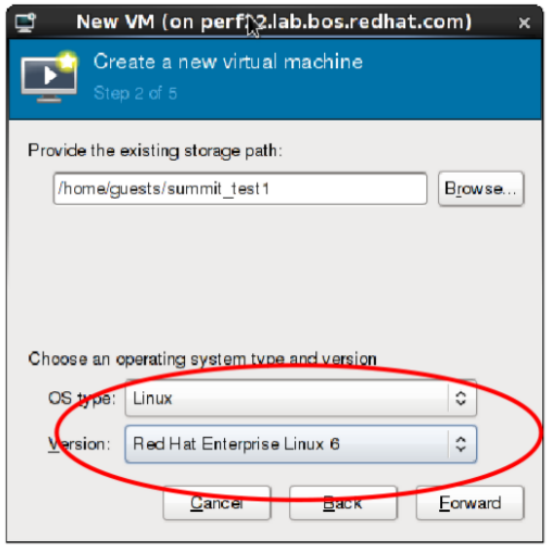
- Los pools son orígenes de dispositivos de almacenamiento en los que alojar los volúmenes (máquinas virtuales)
- Pueden ser de diversos tipos: dir, disk, fs (dispositivos de bloque), iscsi, logical (lvm)

virt-manager - Clonado de máquinas

- Para clonarlas es necesario que esten apagadas.
- Permite hasta compartir los discos.

virt-manager - Rendimiento





Se específico!

virt-manager - Rendimiento

- Usa drivers paravirtualizados (virtio) en la medida de lo posible.
- Usar kernel $>$ 3.0.0 si es posible

Virtualización en Linux - virsh, línea de órdenes de administración

virsh, la shell - Conceptos

- Virsh es una herramienta de línea de comando para administar a los huéspedes y al hipervisor
- Utiliza la API de libvirt
- "virsh -c qemu+ssh://root@maquinakvm/system"

Table 2: Virsh Commands at a Glance

Command	Function
`list --all`	Lists all virtual machines.
`list --inactive`	Lists all inactive virtual machines.
`dominfo VM`	Returns information about the virtual machine named *VM*.
`nodeinfo`	Returns information about the guest system.
`edit VM`	Changes the settings for the named virtual machine.
`start VM`	Starts the named virtual machine.
`shutdown VM`	Shuts down the named virtual machine.
`destroy VM`	Kills the named virtual machine.
`suspend VM`	Pauses the named virtual machine.
`resume VM`	Resumes the named virtual machine.
`console VM`	Activates the management console for the named virtual machine.
`dumpxml VM`	Sends the XML configuration file of the named virtual machine to standard output.
`create vm.xml`	Creates a new virtual machine from information in the `vm.xml` configuration file.
`undefine VM`	Deletes the entire named virtual machine.
`setmem VM memory`	Allots *mem* kilobytes of memory to the named virtual machine.
`setvcpus VM cpu`	Assigns *cpu* virtual CPUs to the named virtual machine.
`migrate VM URI`	Migrates the named virtual machine to the *URI* guest system.

virsh, la shell - Backup

- virsh suspend maquina; cp disco.qcow2 destino; virsh resume maquia

virsh, la shell - Migraciones de máquinas

- Las migraciones p2v requieren realizar un proceso de empaquetamiento del servidor físico en un fichero virtual. Herramienta dd
- `http://libguestfs.org/virt-p2v.1.html`
- Las migraciones v2v requieren básicamente que las máquinas virtuales sean convertidas al formato KVM.
- `http://libguestfs.org/virt-v2v.1.html`
- `https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/6/html-single/V2V_Guide/index.html`
- qemu-img
  - qemu-img convert -f vmdk disco-origen.vmdk -O qcow2 disco-destino.qcow2
  - qemu-img resize disco.qcow2 +50G (+ añade, − quita y nada valor total)
- Soporta una gran cantidad de formatos: raw, qcow2, vdi, vmdk

virsh, la shell - qemu-img

- permite chequear la imagen en disco
- permite realizar conversiones entre formatos
- permite redimensionar una imagen

virsh, la shell - snapshot

- permite realizar snapshot de las máquinas, siempre que el disco sea qcow2
- crear una snapshot, "virsh snapshot-create vm1"
- listar snapshot, "virsh snapshot-list vm1"
- restaurar un snapshot, "virsh snapshot-rever vm1 estadoinicial"
- eliminar un snapshot y continuar, "virsh snapshot-delete vm1 estadoaborrar"

Servicios básicos de red

Servicios básicos de red - Espacios de almacenamiento compartido

Nos basaremos principalmente en la distribución Zentyal