

Linux 2

ITINF 2021

Lektion 9

Idag

- Virtuella servrar (2/2)
- KVM
- Bygga miljö med virtuella servrar

Virtuella servrar

- Bygga upp ett antal virtuella servrar på en fysisk server
- I princip hela serverparken på en maskin
- En hel testmiljö på en maskin, spridd över olika servrar
- ...eller en av flera fysiska maskiner som används för en uppsättning virtuella servrar

KVM

“ ”**KVM** (for **Kernel-based Virtual Machine**) is a full virtualization solution for Linux on x86 hardware containing virtualization extensions (Intel VT or AMD-V). It consists of a loadable kernel module, `kvm.ko`, that provides the core virtualization infrastructure and a processor specific module, `kvm-intel.ko` or `kvm-amd.ko`.

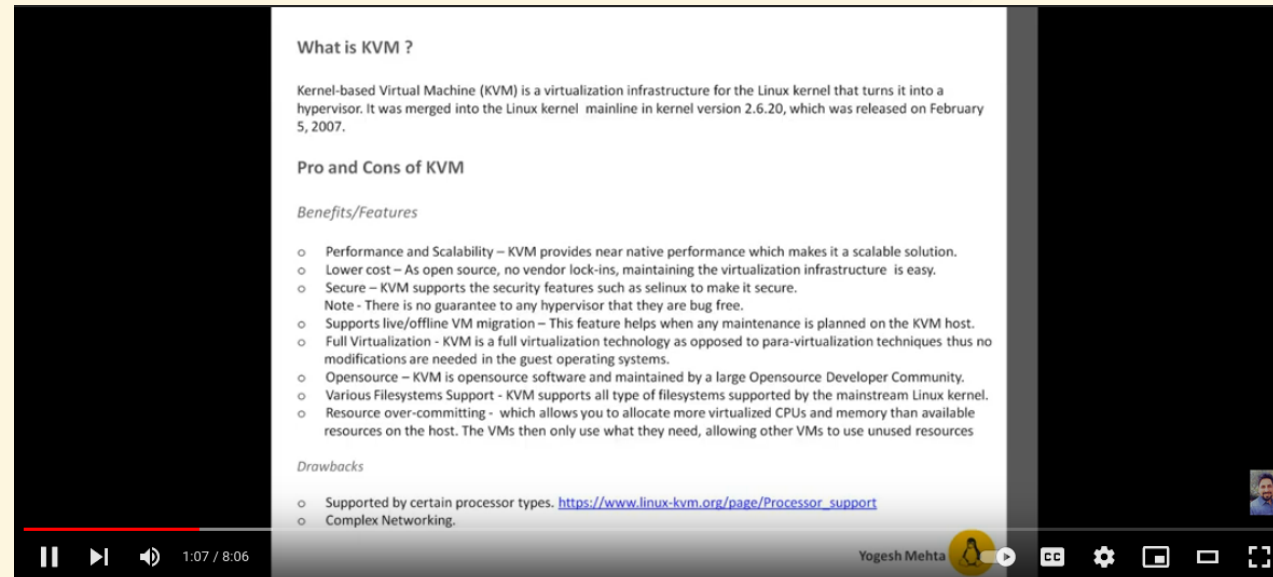
Using KVM, one can run multiple virtual machines running unmodified Linux or Windows images. Each virtual machine has private virtualized hardware: a network card, disk, graphics adapter, etc.

KVM is open source software. The kernel component of KVM is included in mainline Linux, as of 2.6.20. The userspace component of KVM is included in mainline QEMU, as of 1.3.”

KVM

- En hypervisor inbyggd i Linux
- Program på Linux
- I de virtuella maskinerna kan olika operativsystem installeras

KVM



Kort introduktion på YouTube:

<https://www.youtube.com/watch?v=Pgltb5InnLY>

KVM

- `qemu` för att sköta kontakten med värdmaskinens hårdvara
- `virsh` för kommandoraden
- `virt-manager` för GUI

KVM

- Kvm check

- Paket `cpu-checker`
- Kommando `kvm-ok`

- Installera paket

- ```
sudo apt-get install qemu-kvm libvirt-daemon-system libvirt-clients bridge-utils virtinst
```

- `virsh`

# Övning 1

- Kör `kvm-ok`
  - Saknar du det paketet, `apt install cpu-checker`
- Kan din maskin köra KVM?

# Förberedelser

- Om du inte råkar ha någon iso-image liggande på din hårddisk, börja ladda hem en nu så att den finns redo om en stund. (Använd t ex `wget`.)
- Några relativt små images:
  - CentOS8 minimal  
[http://ftp.lysator.liu.se/pub/CentOS/7.9.2009/isos/x86\\_64/CentOS-7-x86\\_64-Minimal-2009.iso](http://ftp.lysator.liu.se/pub/CentOS/7.9.2009/isos/x86_64/CentOS-7-x86_64-Minimal-2009.iso)
  - TinyCore <http://tinycorelinux.net/11.x/x86/release/Core-current.iso>
  - Plain debian <https://cdimage.debian.org/debian-cd/current/amd64/iso-cd/debian-11.4.0-amd64-netinst.iso>

# virsh-kommandon

- `virsh list`
- `virsh list --all`
- `virsh net-list`
- `virsh net-info default`
- `virsh nodeinfo`

# KVM-bibliotek

- `/var/lib/libvirt/`
- ISO images: `/var/lib/libvirt/boot/`
- VM-installationer: `/var/lib/libvirt/images/`
- Libvirt-konfiguration: `/etc/libvirt/`

# Övning 2

- Installera paketen för att använda KVM
- Se till att `virsh list` ger en (tom) lista
- Se till att `virsh net-list` ger en (tom) lista
- Se nätverksinformation med `virsh net-info default`

# Övning 2

```
mo@sputnik:~$ kvm-ok
INFO: /dev/kvm exists
KVM acceleration can be used
mo@sputnik:~$
mo@sputnik:~$ virsh list
 Id Name State

```

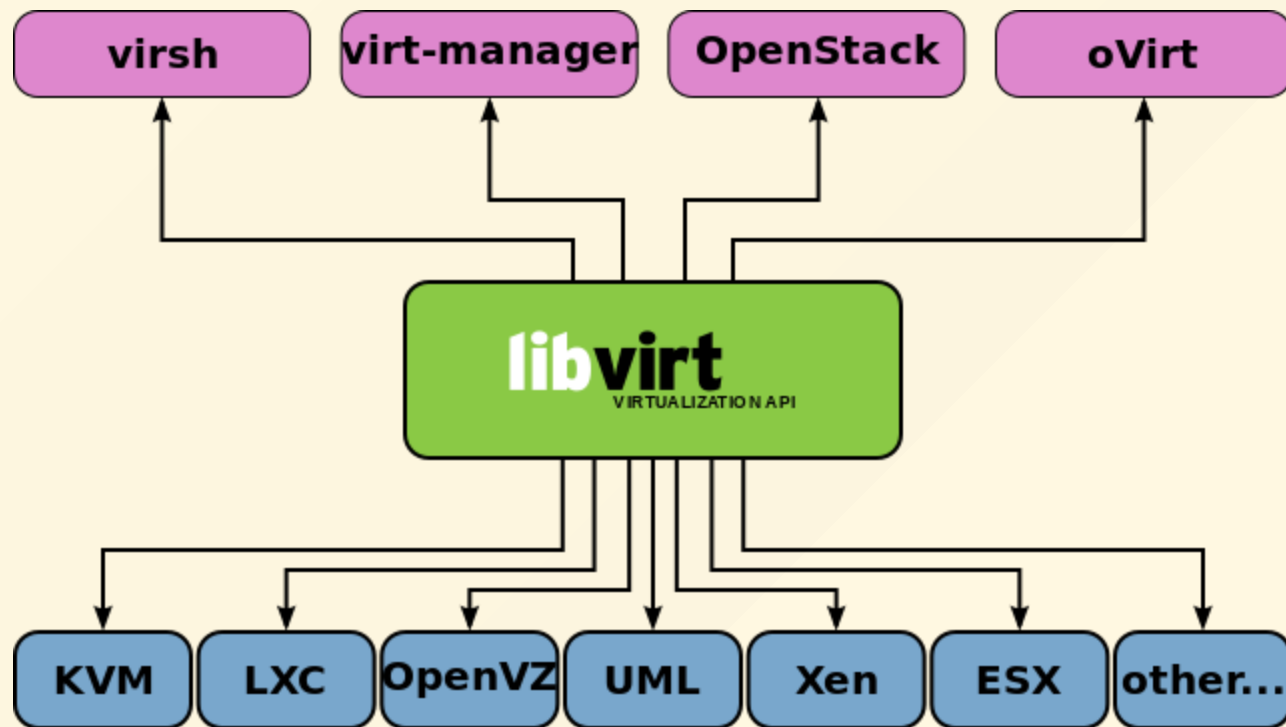
| Name    | State  | Autostart | Persistent |
|---------|--------|-----------|------------|
| default | active | yes       | yes        |

```
mo@sputnik:~$ virsh net-list
 Name State Autostart Persistent

 default active yes yes

mo@sputnik:~$ virsh net-info default
Name: default
UUID: 2985b4e5-d9cb-49d2-81bb-f927fb89ba51
Active: yes
Persistent: yes
Autostart: yes
Bridge: virbr0
```

# Libvirt är en del av andra lösningar också





KVM

# Bygg en VM

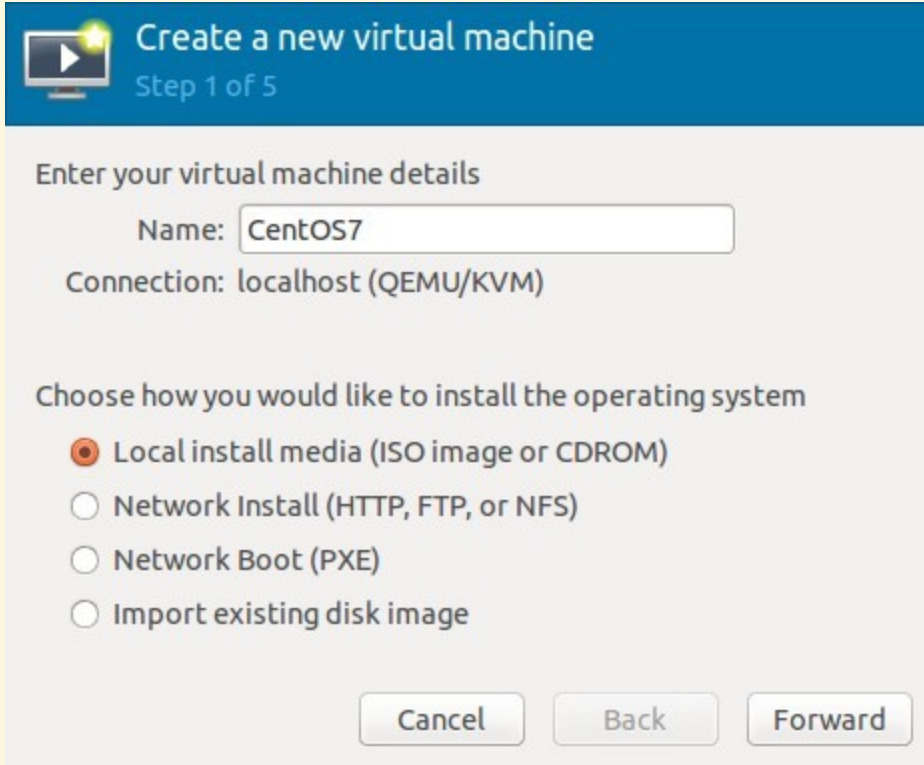
# KVM: bygga en VM

- Börja med att hämta en iso, lämpligen för någon minimerad Linux och kopiera den till `/var/lib/libvirt/boot/`

```
cd /var/lib/libvirt/boot/
sudo wget http://ftp.lysator.liu.se/pub/CentOS/7.9.2009/isos/x86_64/CentOS-7-x86_64-Minimal-2009.iso
```

- Installera med virt-manager (GUI) eller virt-install (CLI)

# virt-manager



The image shows a screenshot of the 'Create a new virtual machine' dialog box in the virt-manager application. The dialog has a blue header bar with a play button icon and the text 'Create a new virtual machine' and 'Step 1 of 5'. Below the header, the text 'Enter your virtual machine details' is followed by a 'Name:' label and a text input field containing 'CentOS7'. Below this, the text 'Connection: localhost (QEMU/KVM)' is displayed. The next section is titled 'Choose how you would like to install the operating system' and contains four radio button options: 'Local install media (ISO image or CDROM)' (which is selected), 'Network Install (HTTP, FTP, or NFS)', 'Network Boot (PXE)', and 'Import existing disk image'. At the bottom of the dialog are three buttons: 'Cancel', 'Back', and 'Forward'.

Create a new virtual machine  
Step 1 of 5

Enter your virtual machine details

Name:

Connection: localhost (QEMU/KVM)

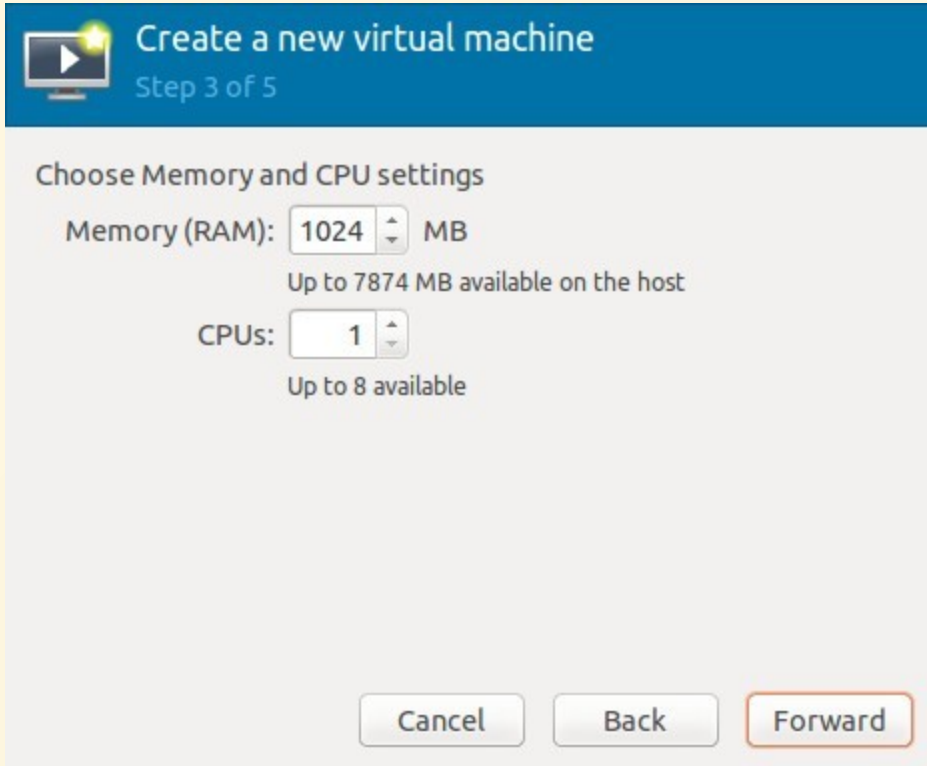
Choose how you would like to install the operating system

- ☒ Local install media (ISO image or CDROM)
- ☐ Network Install (HTTP, FTP, or NFS)
- ☐ Network Boot (PXE)
- ☐ Import existing disk image

# virt-manager



# virt-manager



Create a new virtual machine  
Step 3 of 5

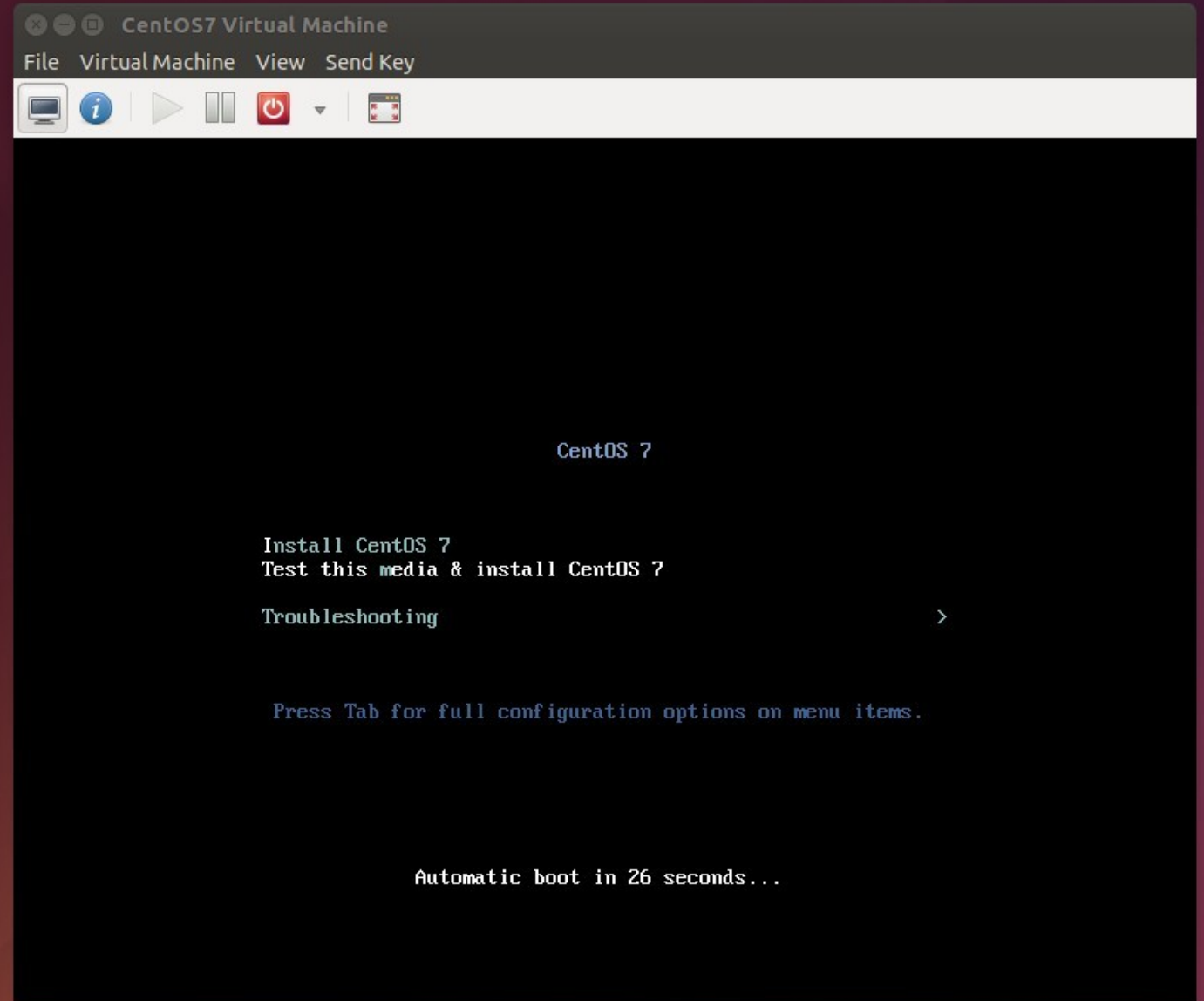
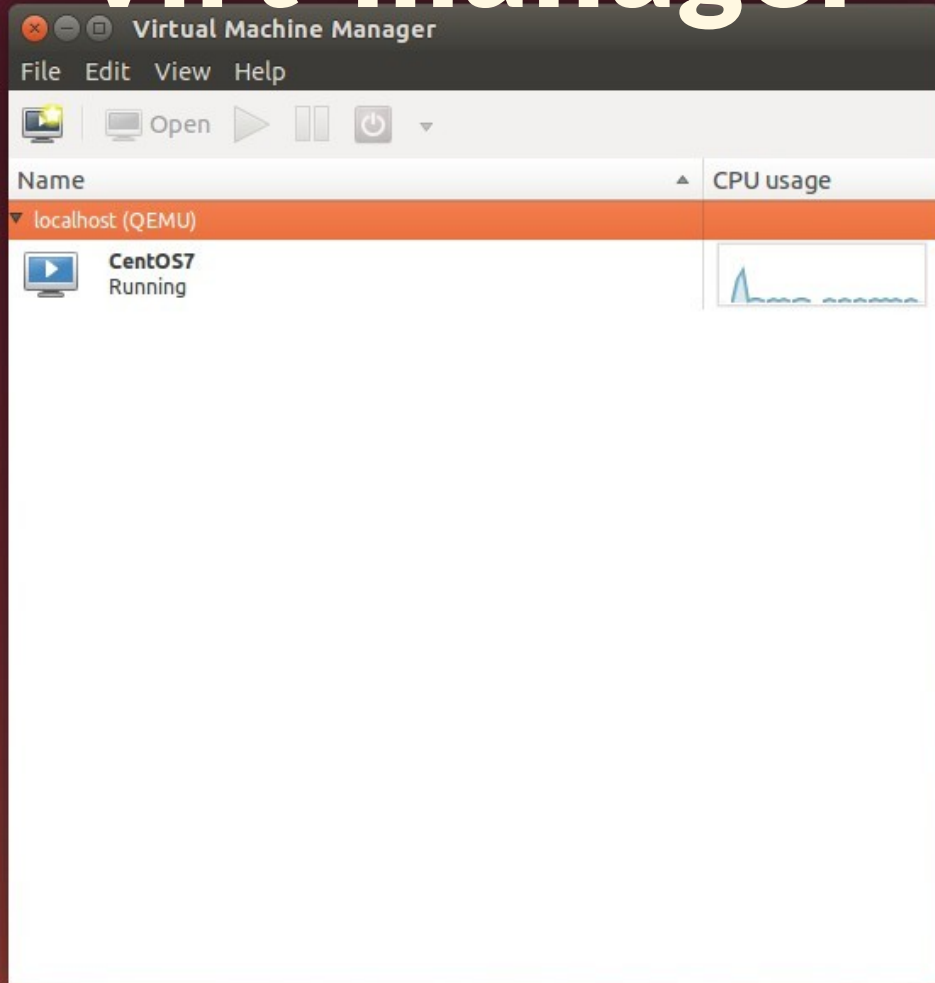
Choose Memory and CPU settings

Memory (RAM): 1024 MB  
Up to 7874 MB available on the host

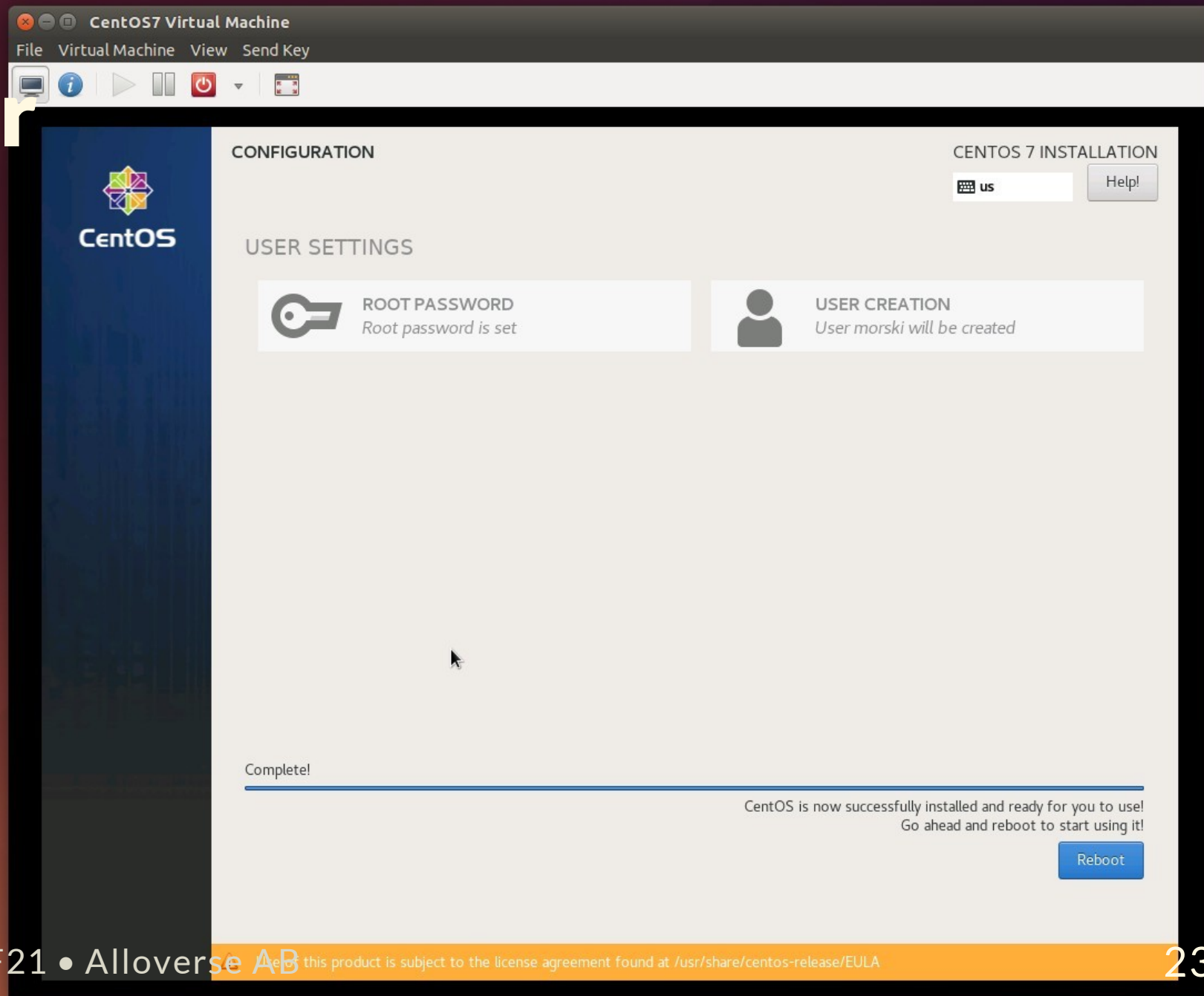
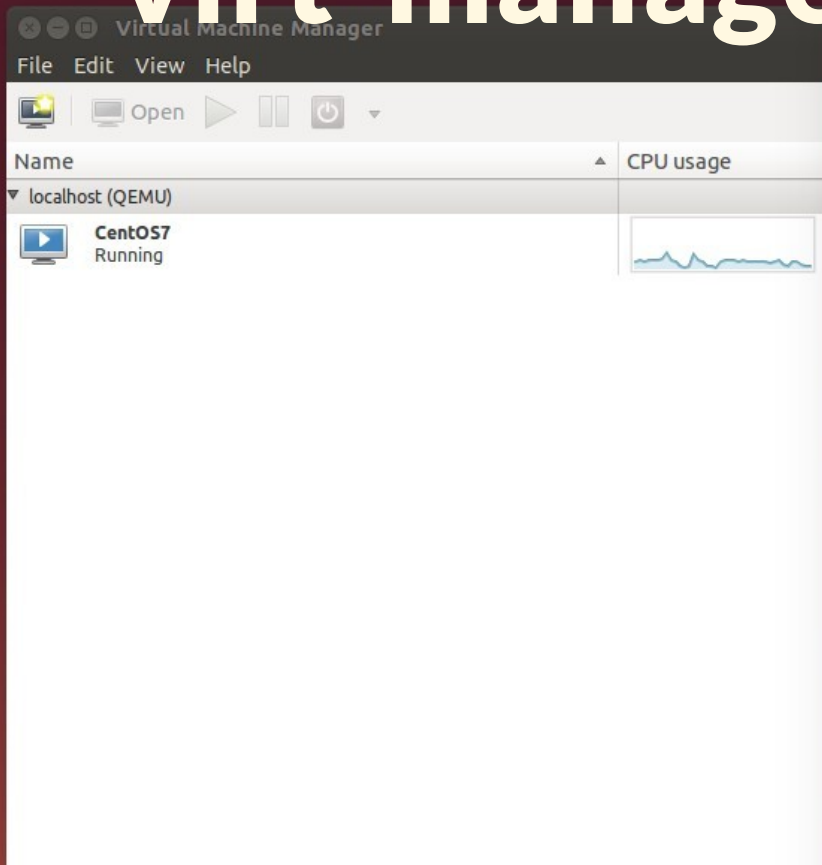
CPUs: 1  
Up to 8 available

Cancel Back Forward

# virt-manager



# virt-manager



# virt-install, exempel

```
sudo virt-install \
 --name centos7 \
 --description "Test VM with CentOS 7" \
 --ram=1024 \
 --vcpus=2 \
 --os-type=Linux \
 --os-variant=rhel7.0 \
 --disk path=/var/lib/libvirt/images/centos7.qcow2,bus=virtio,size=10 \
 --graphics none \
 --location /var/lib/libvirt/boot/CentOS-7-x86_64-Minimal-2009.iso \
 --network bridge:virbr0 \
 --console pty,target_type=serial \
 -x 'console=ttyS0,115200n8 serial'
```

från <https://computingforgeeks.com/virsh-commands-cheatsheet/>



```
nevy@nevy-linux2: ~/Dev/nackademin-linux2
Starting installer, one moment...
anaconda 21.48.22.159-1 for CentOS 7 started.
* installation log files are stored in /tmp during the installation
* shell is available on TTY2
* when reporting a bug add logs from /tmp as separate text/plain attachments
13:05:57 Not asking for VNC because we don't have a network
=====
Installation

1) [x] Language settings 2) [!] Time settings
 (English (United States)) (Timezone is not set.)
3) [!] Installation source 4) [!] Software selection
 (Processing...) (Processing...)
5) [!] Installation Destination 6) [x] Kdump
 (No disks selected) (Kdump is enabled)
7) [] Network configuration 8) [!] Root password
 (Not connected) (Password is not set.)
9) [!] User creation
 (No user will be created)

Please make your choice from above ['q' to quit | 'b' to begin installation |
'r' to refresh]:
```

# Fler virsh-kommandon

- `virsh start <vm>`
- `virsh shutdown <vm>`
- `virsh destroy <vm>`
- `virsh reboot <vm>`
- `virsh dominfo <vm>`
- `virsh undefine <vm>`

# Övning 3

- Installera en virtuell maskin på KVM.
- Kontrollera att du kan starta och starta om din virtuella maskin.

# Övning 3

- `virsh start CentOS7`
- `virsh shutdown CentOS7`
- `virsh start CentOS7`
- `virsh reboot CentOS7`
- `virsh dominfo CentOS7`

KVM

# Klona, flytta VM

# virt-clone

- "Clone", skapa en kopia av en virtuell server med `virt-clone`
- Maskinen som skall klonas behöver vara avstängd under kloningen
- Syntax:

```
virt-clone --original {Domain-Vm-Name-Here} --name {New-Domain-Vm-Name-Here} --auto-clone
```

# Övning 4

Klona den virtuella maskin som du skapade i övning 3

# Övning 4: Lösning

```
$ sudo virt-clone --original CentOS7 --name CentOS7-2 --auto-clone
$ virsh start CentOS7-2
```



# Kopiera till annan fysisk server

Inte en lika automatisk kloning som lokalt, men med enkla steg:

1. Stäng ned VMen som skall kopieras.
2. Kopiera image-filen från `/var/lib/libvirt/images` på den enba maskinen till den andra
3. Plocka ut vm-definitionerna till en XML-fil som du kopierar till den ny aservern

```
virsh dumpxml VMNAME > domxml.xml
```

4. Läs in vm-definitionerna på den nya servern

```
virsh define domxml.xml
```

KVM

# Nätverk

# Nätverk mellan virtuella maskiner

- Default-uppsättning i kvm ger varje virtuell maskin en adress på nätet `192.168.122.0/24`
- Värddmaskinen kan nås på `192.168.122.1` (din VMs gateway)
- Notera att detta gäller för nätverksuppsättningen `hypervisor default`
- Notera NAT-inställningar som har skapats i `iptables`

# Övning 5

- Titta på dina virtuella maskiners nätverksinställningar och hitta vilka adresser de har fått
- Testa förbindelsen mellan dina två virtuella maskiner (förslagsvis med `ping`)
- Testa förbindelsen till `192.168.122.1`
- Ta en titt på vad som hänt i `iptables` (på värdmaskinen)
- Ta en titt på nätverksinterface `vibr0` (på värdmaskinen)

# Övning 5

```
mo@sputnik:~/nackademin/Linux2$ sudo iptables -n -t nat -L
Chain PREROUTING (policy ACCEPT)
target prot opt source destination

Chain INPUT (policy ACCEPT)
target prot opt source destination

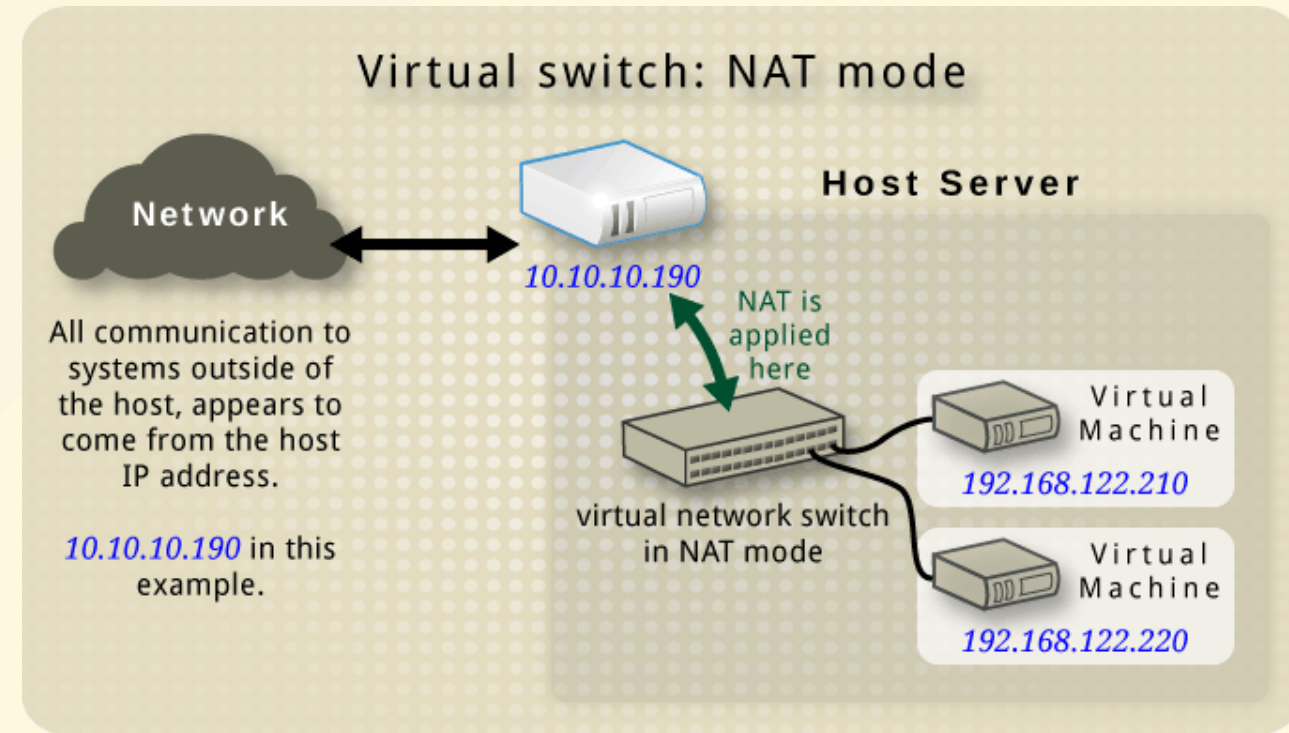
Chain OUTPUT (policy ACCEPT)
target prot opt source destination

Chain POSTROUTING (policy ACCEPT)
target prot opt source destination
RETURN all -- 192.168.122.0/24 224.0.0.0/24
RETURN all -- 192.168.122.0/24 255.255.255.255
MASQUERADE tcp -- 192.168.122.0/24 !192.168.122.0/24 masq ports: 1024-65535
MASQUERADE udp -- 192.168.122.0/24 !192.168.122.0/24 masq ports: 1024-65535
MASQUERADE all -- 192.168.122.0/24 !192.168.122.0/24
mo@sputnik:~/nackademin/Linux2$
```

```
virbr0 Link encap:Ethernet HWaddr fe:54:00:14:0a:96
 inet addr:192.168.122.1 Bcast:192.168.122.255 Mask:255.255.255.0
 UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
 RX packets:38 errors:0 dropped:0 overruns:0 frame:0
 TX packets:27 errors:0 dropped:0 overruns:0 carrier:0
 collisions:0 txqueuelen:0
 RX bytes:3466 (3.4 KB) TX bytes:3029 (3.0 KB)
```

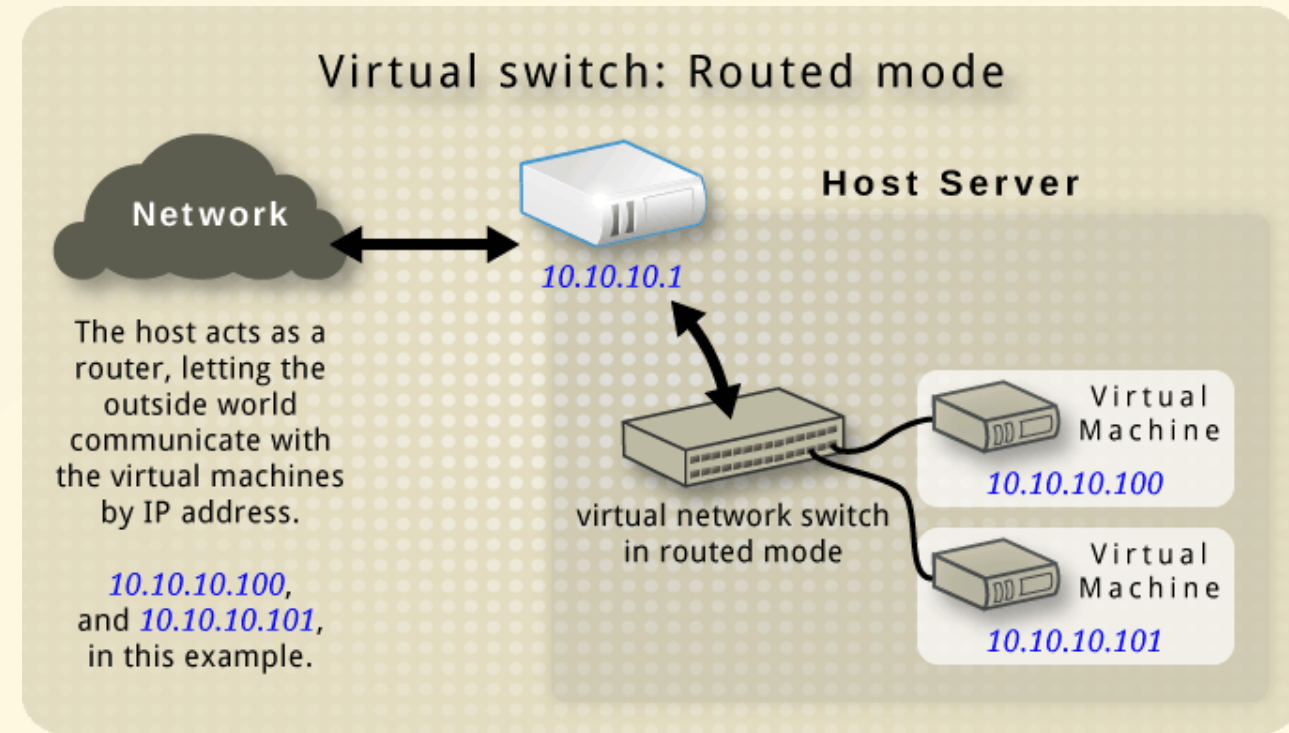
# Nätverksaccess: NAT

- I `default` så agerar hosten NAT-router, och VMs kan inte nås från världen
- <https://wiki.libvirt.org/page/VirtualNetworking>



# Nätverksaccess: Routed

- I "routed" mode skapar hosten ett nätverk, men utan NAT, så yttre nätverket kan konfigureras så trafik kan nå VMs (men yttre router måste confas)



# Nätverksaccess: Bridged

- Skapa en brygga (bridge) för att göra virtuella maskiner tillgängliga genom nätverket
- Använder sig av värdmaskinens nätverk
- Fungerar inte med wlan
- Skulle kunna `ip link add`... men det är inte persistent
- old-school var `/etc/network/interfaces`, men nuförtiden används netplan
- Outdated men lättläst: <https://linuxconfig.org/how-to-use-bridged-networking-with-libvirt-and-kvm>
- Modern men svårläst: <https://www.answertopia.com/ubuntu/creating-an-ubuntu-kvm-networked-bridge-interface/>



# Nätverksaccess: Bridged

```
$ networkctl status -a # kolla vad ditt nätverkskort heter
$ pico /etc/netplan/01-network-manager-all.yaml # definera bryggan
network:
 version: 2
 renderer: NetworkManager
 ethernets:
 enp4s0:
 dhcp4: true
 bridges:
 br0:
 interfaces: [enp4s0]
 dhcp4: yes
$ sudo netplan apply # starta om networking
```

# Nätverksaccess: Bridged

```
$ nmcli con add ifname br0 type bridge con-name br0
koppla bryggan till fysiska nätverks-kortet
$ nmcli con add type bridge-slave ifname enp4s0 master br0
byt nätverkskort från fysiska till bryggan
$ nmcli con down netplan-enp4s0
$ nmcli con up br0
skapa kortet i libvirt
$ cat > bridge.xml
<network>
 <name>br0</name>
 <forward mode="bridge"/>
 <bridge name="br0" />
</network>
```

# Nätverksaccess: Bridged

```
$ virsh net-define bridge.xml
$ virsh net-start br0
$ virsh net-autostart br0
... och sen använd `br0` istället för `default` på din virtuella maskin.
$ virsh edit centos8
 <interface type='network'>
 <mac address='52:54:00:e6:a9:57' />
 <source network='br0' />
 <model type='virtio' />
 <address type='pci' domain='0x0000' bus='0x01' slot='0x00' function='0x0' />
 </interface>
```

Tillbakablick, reflektion, kommentarer ...