# Qemu dependencies

QEMU is packaged by most Linux distributions:

* **Arch:** pacman -S qemu
* **Debian/Ubuntu:** apt-get install qemu
* **Fedora:** dnf install @virtualization
* **Gentoo:** emerge --ask app-emulation/qemu
* **RHEL/CentOS:** yum install qemu-kvm
* **SUSE:** zypper install qemu
* QEMU can be installed from **Homebrew**:
* brew install qemu
* QEMU can be installed from **MacPorts**:
* sudo port install qemu

# Install Qemu on Ubuntu 20.04

**$ qemu-system-x86\_64 --version**

Command 'qemu-system-x86\_64' not found, …

**$ $?**

127: command not found

**$ sudo apt-get update -y**

[sudo] password for laurent:

Hit:1 http://fr.archive.ubuntu.com/ubuntu focal InRelease

…

**$ sudo apt install -y qemu qemu-utils qemu-kvm**

…

The following NEW packages will be installed:

cpu-checker ibverbs-providers ipxe-qemu ipxe-qemu-256k-compat-efi-roms libaio1 libcacard0 libfdt1 libibverbs1

libiscsi7 libpmem1 librados2 librbd1 librdmacm1 libslirp0 libspice-server1 libusbredirparser1 libvirglrenderer1

msr-tools ovmf qemu-block-extra qemu-kvm qemu-system-common qemu-system-data qemu-system-gui qemu-system-x86

qemu-utils seabios sharutils

0 upgraded, 28 newly installed, 0 to remove and 189 not upgraded.

Need to get 19,7 MB of archives.

After this operation, 82,6 MB of additional disk space will be used.

Do you want to continue? [Y/n] y …

Processing triggers for man-db (2.9.1-1) ...

**$ kvm-ok (**apt-get install cpu-checker**]**

INFO: /dev/kvm exists

KVM acceleration can be used

**$ qemu-system-x86\_64 --version**

QEMU emulator version 4.2.1 (Debian 1:4.2-3ubuntu6.19)

Copyright (c) 2003-2019 Fabrice Bellard and the QEMU Project developers

**$ $?**

0: command not found

# Create user ssh key

**$ ls ~/.ssh/id\_rsa**

**$ ssh-keygen**

Generating public/private rsa key pair.

Enter file in which to save the key (/home/laurent/.ssh/id\_rsa):

Enter passphrase (empty for no passphrase):

Enter same passphrase again:

Your identification has been saved in /home/laurent/.ssh/id\_rsa

Your public key has been saved in /home/laurent/.ssh/id\_rsa.pub

The key fingerprint is:

SHA256:Um9X7e2VaLP/EFSqqMrm9o9ZAVocB1NmJCXADVlAnhA laurent@ubuntu-XPS-8700

# Start an Ubuntu cloud VM

**$ mkdir vm**

**Exit**

$ scp config.toml.tmpl [laurent@192.168.1.29:~/vm](mailto:laurent@192.168.1.29:~/vm)

$ scp helm [laurent@192.168.1.29:~/vm](mailto:laurent@192.168.1.29:~/vm)

$ scp k3s [laurent@192.168.1.29:~/vm](mailto:laurent@192.168.1.29:~/vm)

$ scp k3s-airgap-images-amd64.tar [laurent@192.168.1.29:~/vm](mailto:laurent@192.168.1.29:~/vm)

**ssh**

**$ cd vm**

**$ wget https://cloud-images.ubuntu.com/minimal/releases/focal/release-20220201/ubuntu-20.04-minimal-cloudimg-amd64.img**

Connecting to cloud-images.ubuntu.com (cloud-images.ubuntu.com)|2001:67c:1360:8001::34|:443... connected.

HTTP request sent, awaiting response... 200 OK

Length: 258473984 (246M)

Saving to: ‘ubuntu-20.04-minimal-cloudimg-amd64.img’

**$ cat > metadata.yaml <<EOF**

**instance-id: iid-wordslab**

**local-hostname: wordslab**

**EOF**

**$ cat > user-data.yaml <<EOF**

**#cloud-config**

**ssh\_authorized\_keys:**

**- $(cat ~/.ssh/id\_rsa.pub)**

**EOF**

**$ cat > user-data.yaml <<EOF**

**#cloud-config**

**ssh\_authorized\_keys:**

**- $(cat ~/.ssh/id\_rsa.pub)**

**device\_aliases:**

**wordslab-cluster: /dev/vdc  
 wordslab-data: /dev/vdd**

**disk\_setup:**

**wordslab-cluster:**

**table\_type: gpt**

**layout: true**

**wordslab-data:**

**table\_type: gpt**

**layout: true**

**fs\_setup:**

**- label: wordslab-cluster**

**device: wordslab-cluster.1  
 filesystem: ext4**

**- label: wordslab-data**

**device: wordslab-data.1**

**filesystem: ext4**

**bootcmd:**

**- mkdir -p /mnt/wordslab-cluster**

**- mkdir -p /mnt/wordslab-data**

**- mkdir -p /etc/rancher**

**- mkdir -p /var/lib**

**- mkdir -p /var/log/rancher**

**- mkdir -p /var/volume/rancher**

**mounts:**

**- ["wordslab-cluster.1", "/mnt/wordslab-cluster", "ext4", "defaults", "0", "2"]**

**- ["wordslab-data.1", "/mnt/wordslab-data", "ext4", "defaults", "0", "2"]**

**packages:**

**- vim-tiny**

**runcmd:**

**- mkdir -p /mnt/wordslab-cluster/etc/rancher/k3s**

**- mkdir -p /mnt/wordslab-cluster/var/lib/rancher/k3s**

**- mkdir -p /mnt/wordslab-cluster/var/log/rancher/k3s**

**- mkdir -p /mnt/wordslab-data/var/volume/rancher/k3s**

**EOF**

mnt/wordslab-cluster ext4 defaults 0 2

**$ sudo apt-get install -y cloud-image-utils**

The following NEW packages will be installed:

cloud-image-utils

0 upgraded, 1 newly installed, 0 to remove and 189 not upgraded.

Need to get 16,9 kB of archives.

After this operation, 62,5 kB of additional disk space will be used**.**

**$ cloud-localds seed.img user-data.yaml metadata.yaml**

**$ qemu-img create -b ubuntu-20.04-minimal-cloudimg-amd64.img -f qcow2 wordslab-os.img**

**$ qemu-img create -f qcow2 wordslab-cluster.img 10G**

**$ qemu-img create -f qcow2 wordslab-data.img 10G**

#NOTE : How to access the images from the host OS  
$ sudo modprobe nbd

**$** sudo qemu-nbd -c /dev/nbd0 **wordslab-cluster.img**

**$ sudo mkdir -p /mnt/wordslab-cluster  
$ sudo mount /dev/nbd0p1 /mnt/wordslab-cluster**

**$** sudo qemu-nbd -c /dev/nbd1 **wordslab-os.img**

**$ sudo mkdir -p /mnt/wordslab-os  
$ sudo mount /dev/nbd1p1 /mnt/wordslab-os**

**$ qemu-system-x86\_64 \**

**-machine accel=kvm,type=q35 \**

**-cpu host \**

**-smp $(nproc) \**

**-m 8G \**

**-nographic \**

**-device virtio-net-pci,netdev=net0 \**

**-netdev user,id=net0,hostfwd=tcp::3022-:22,hostfwd=tcp::3080-:80,hostfwd=tcp::3443-:6443 \**

**-drive if=virtio,format=qcow2,file=wordslab-os.img \**

**-drive if=virtio,format=raw,file=seed.img \  
 -drive if=virtio,format=qcow2,file=wordslab-cluster.img \  
 -drive if=virtio,format=qcow2,file=wordslab-data.img**

…

**$ ssh-keyscan -H -p 3022 0.0.0.0 >> ~/.ssh/known\_hosts**

**$ ssh -p 3022** [**ubuntu@0.0.0.0**](mailto:ubuntu@0.0.0.0)

Welcome to Ubuntu 20.04.3 LTS (GNU/Linux 5.4.0-1054-kvm x86\_64)

**…**

**$ scp -P 3022 k3s ubuntu@0.0.0.0:**~/k3s

**scp -P 3022 k3s-airgap-images-amd64.tar ubuntu@0.0.0.0:**~/k3s-airgap-images-amd64.tar

**scp -P 3022 helm ubuntu@0.0.0.0:**~/helm

ubuntu@wordslab:~$

**- mount -o bind /mnt/wordslab-cluster/etc/rancher /etc/rancher**

**- mount -o bind /mnt/wordslab-cluster/var/lib /var/lib**

**- mount -o bind /mnt/wordslab-cluster/var/log/rancher /var/log/rancher**

**- mount -o bind /mnt/wordslab-data/var/volume/rancher /var/volume/rancher**

**ssh -p 3022 ubuntu@0.0.0.0 chmod a+x wordslab-k3s\*.sh**

**ssh -p 3022 ubuntu@0.0.0.0 sudo ./wordslab-k3s-install.sh**

**$ sudo bash -c "echo 'root:password' | chpasswd"**

**$ su root**

mkdir -p /mnt/wordslab-cluster/etc/rancher

mkdir -p /mnt/wordslab-cluster/var/lib

mkdir -p /mnt/wordslab-cluster/var/log/rancher

mkdir -p /mnt/wordslab-data/var/volume/rancher/k3s

sudo cp k3s /mnt/wordslab-os/usr/local/bin

cp $downloadpath/$2 /usr/local/bin/k3s

chmod a+x /usr/local/bin/k3s

echo -e "alias kubectl='k3s kubectl'" >> ~/.bash\_aliases

source ~/.bashrc

sudo cp helm /mnt/wordslab-os/usr/local/bin

cp $downloadpath/$3 /usr/local/bin/helm

chmod a+x /usr/local/bin/helm

mkdir -p /mnt/wordslab-cluster/var/lib/rancher/k3s/agent/images

sudo cp k3s-airgap-images-amd64.tar /mnt/wordslab-cluster/var/lib/rancher/k3s/agent/images

cp $downloadpath/$2 /mnt/wordslab-cluster/var/lib/rancher/k3s/agent/images/k3s-airgap-images-amd64.tar

scp -p 3022 k3s-airgap-images-amd64.tar ubuntu@0.0.0.0:/mnt/wordslab-cluster/var/lib/rancher/k3s/agent/images

qemu-nbd -d wordslab-os.img

qemu-nbd -d wordslab-os.img => still locked

[Modules — cloud-init 21.4 documentation (cloudinit.readthedocs.io)](https://cloudinit.readthedocs.io/en/latest/topics/modules.html)

---

mkdir -p /etc/rancher/k3s

rm -f /etc/rancher/k3s/k3s.yaml

mkdir -p /var/lib/rancher/k3s

k3s --version | grep -o "v[0-9].\*\s" > /var/lib/rancher/k3s/version

mkdir -p /var/log/rancher/k3s

mkdir -p /var/volume/rancher/k3s

nohup /usr/local/bin/k3s server --https-listen-port 6443 --log /var/log/rancher/k3s/k3s-$(date +%Y%m%d-%H%M%S).log --default-local-storage-path /var/volume/rancher/k3s </dev/null >/dev/null 2>&1 &

sleep 1

ps | grep -o "[0-9].\*k3s\sserver" | grep -Eo "^[0-9]+" > /var/lib/rancher/k3s/pid

# K3s version : cat /var/lib/rancher/k3s/version

# K3s process id : cat /var/lib/rancher/k3s/pid

# VM IP address : hostname -I | grep -Eo "^[0-9\.]+"

# Kubeconfig file: cat /etc/rancher/k3s/k3s.yaml

---

Exit qemu : Crtl-A x