

# How to Install and Configure KVM on Ubuntu 18.04 LTS

**Original How-To URL:** <https://www.linuxtechi.com/install-configure-kvm-ubuntu-18-04-server/>

**KVM** (Kernel-based Virtual Machine) is an open source full virtualization solution for Linux like systems, KVM provides virtualization functionality using the virtualization extensions like **Intel VT** or **AMD-V**. Whenever we install KVM on any linux box then it turns it into the hypervisor by loading the kernel modules like **kvm-intel.ko**( for intel based machines) and **kvm-amd.ko** ( for amd based machines).

KVM allows us to install and run multiple virtual machines (Windows & Linux). We can create and manage KVM based virtual machines either via **virt-manager** graphical user interface or **virt-install** & **virsh** cli commands.

In this article we will discuss how to install and configure **KVM hypervisor** on Ubuntu 18.04 LTS server. I am assuming you have already installed Ubuntu 18.04 LTS server on your system. Login to your server and perform the following steps.

## Verify Whether your system support hardware virtualization

Execute below egrep command to verify whether your system supports hardware virtualization or not,

```
egrep -c '(vmx|svm)' /proc/cpuinfo
```

```
1
```

If the output is greater than 0 then it means your system supports Virtualization else reboot your system, then go to BIOS settings and enable VT technology.

Now Install “**kvm-ok**” utility using below command, it is used to determine if your server is capable of running hardware accelerated KVM virtual machines

```
sudo apt install cpu-checker
```

Run kvm-ok command and verify the output,

```
sudo kvm-ok
```

```
INFO: /dev/kvm exists
```

KVM acceleration can be used

## Install KVM and its required packages

Run the below apt commands to install KVM and its dependencies

```
sudo apt update
```

```
sudo apt install qemu qemu-kvm libvirt-bin bridge-utils virt-manager
```

Once the above packages are installed successfully, then your local user (In my case linuxtechi) will be added to the group libvirtd automatically.

## Start & enable libvirt service

Whenever we install qemu & libvirt packages in Ubuntu 18.04 Server then it will automatically start and enable libvirt service, In case libvirt service is not started and enabled then run beneath commands,

```
sudo service libvirt start
```

```
sudo update-rc.d libvirt enable
```

Now verify the status of libvirt service using below command,

```
service libvirt status
```

Output would be something like below:

```

linuxtech@kvm-ubuntu18-04:~$ service libvirtd status
libvirtd.service - Virtualization daemon
  Loaded: loaded (/lib/systemd/system/libvirtd.service; enabled; vendor preset: enabled)
  Active: active (running) since Sat 2018-05-19 16:32:23 UTC; 11min ago
    Docs: man:libvirtd(8)
           https://libvirt.org
  Main PID: 897 (libvirtd)
    Tasks: 19 (limit: 32768)
   CGroup: /system.slice/libvirtd.service
           └─ 897 /usr/sbin/libvirtd
              └─ 1113 /usr/sbin/dnsmasq --conf-file=/var/lib/libvirt/dnsmasq/default.conf --leasefile-ro --dhcp-script=/usr/lib/libvirt/libv
                 └─ 1114 /usr/sbin/dnsmasq --conf-file=/var/lib/libvirt/dnsmasq/default.conf --leasefile-ro --dhcp-script=/usr/lib/libvirt/libv

May 19 16:32:25 kvm-ubuntu18-04 dnsmasq[1113]: compile time options: IPv6 GNU-getopt DBus i18n IDN DHCP DHCPv6 no-Lua TFTP conntrack ips
May 19 16:32:25 kvm-ubuntu18-04 dnsmasq-dhcp[1113]: DHCP, IP range 192.168.122.2 -- 192.168.122.254, lease time 1h
May 19 16:32:25 kvm-ubuntu18-04 dnsmasq-dhcp[1113]: DHCP, sockets bound exclusively to interface virbr0
May 19 16:32:25 kvm-ubuntu18-04 dnsmasq[1113]: reading /etc/resolv.conf
May 19 16:32:25 kvm-ubuntu18-04 dnsmasq[1113]: using nameserver 127.0.0.53#53
May 19 16:32:25 kvm-ubuntu18-04 dnsmasq[1113]: read /etc/hosts - 7 addresses
May 19 16:32:25 kvm-ubuntu18-04 dnsmasq[1113]: read /var/lib/libvirt/dnsmasq/default.addnhosts - 0 addresses
May 19 16:32:25 kvm-ubuntu18-04 dnsmasq-dhcp[1113]: read /var/lib/libvirt/dnsmasq/default.hostsfile
May 19 16:32:25 kvm-ubuntu18-04 dnsmasq[1113]: reading /etc/resolv.conf
May 19 16:32:25 kvm-ubuntu18-04 dnsmasq[1113]: using nameserver 127.0.0.53#53
lines 1-22/22 (END)

```

# Configure Network Bridge for KVM virtual Machines

Network bridge is required to access the KVM based virtual machines outside the KVM hypervisor or host. In Ubuntu 18.04, network is managed by netplan utility, whenever we freshly installed Ubuntu 18.04 server then netplan file is created under **/etc/netplan/**. In most of the hardware and virtualized environment, netplan file name would be “**50-cloud-init.yaml**” or “**01-netcfg.yaml**”, to configure static IP and bridge, netplan utility will refer this file.

As of now I have already configured the static IP via this file and content of this file is below:

Let’s add the network bridge definition in this file:

```

network:
  ethernets:
    ens33:
      addresses: [192.168.0.51/24]
      gateway4: 192.168.0.1
      nameservers:
        addresses: [192.168.0.1]
      dhcp4: no
      optional: true
  version: 2

```

```
sudo vi /etc/netplan/50-cloud-init.yamlnetwork:
```

**version: 2**

**ethernets:**

**ens33:**

**dhcp4: no**

**dhcp6: no**

**bridges:**

**br0:**

**interfaces: [ens33]**

**dhcp4: no**

**addresses: [192.168.0.51/24]**

**gateway4: 192.168.0.1**

**nameservers:**

**addresses: [192.168.0.1]**

As you can see we have removed the IP address from interface(ens33) and add the same IP to the bridge '**br0**' and also added interface (ens33) to the bridge br0. Apply these changes using below netplan command,

```
sudo netplan apply
```

If you want to see the debug logs then use the below command,

```
sudo netplan --debug apply
```

Now Verify the bridge status using following methods:

```
sudo networkctl status -a
```

```
ifconfig
```

## Creating Virtual machine with virt-manager

There are two ways to create virtual machine:

- virt-manager (GUI utility)
- virt-install command (cli utility)

Start the virt-manager by executing the beneath command,

```
sudo virt-manager
```

- Create a new virtual machine
- Click on forward and select the ISO file, in my case I am using RHEL 7.3 iso file.
- Click on Forward
- In the next couple of windows, you will be prompted to specify the RAM, CPU and disk for the VM.
- Now Specify the Name of the Virtual Machine and network,
- Click on Finish

Now follow the screen instruction and complete the installation,

## Creating Virtual machine from CLI using virt-install command

Use the below virt-install command to create a VM from terminal, it will start the installation in CLI, replace the name of the VM, description, location of ISO file and network bridge as per your setup.

```
sudo virt-install -n DB-Server --description "Test VM for Database" --os-type=Linux --os-variant=rhel7 --ram=1096 --vcpus=1 --disk path=/var/lib/libvirt/images/dbserver.img,bus=virtio,size=10 --network bridge:br0 --graphics none --location /home/linuxtechi/rhel-server-7.3-x86_64-dvd.iso --extra-args console=ttyS0
```

---

Revision #1

Created 17 November 2020 19:03:40 by Dino Edwards

Updated 17 November 2020 19:30:41 by Dino Edwards