How to Install KVM on Rocky Linux 9 / AlmaLinux 9

Published on: February 6, 2024 by Neville Ondara

This tutorial will walk you through how to install KVM on Rocky Linux 9 or AlmaLinux 9.

Kernel Virtual Machine (KVM) is an open-source Type1/bare-metal hypervisor that enables users to host and run multiple isolated virtual environments on their Linux machine.

Like most virtualization solutions, KVM abstracts hardware resources, including CPU, memory, storage, network, and graphics, and assigns them to guest machines that run in an isolated environment.

KVM provides numerous functionalities such as resource control, scheduling, scalability, high performance, low latency, live migration, memory management, and more. In addition, you can use Ansible and other automation tools to automate KVM deployments

*Name**

*Email Address**

Subscribe

- Internet Connection
- Basic understanding of Linux networking and commands.

Without any delay, let's deep dive into KVM installations steps.

1. Confirm Hardware Virtualization

Ensure that your system has hardware virtualization extensions enabled. For Intel-based hosts, Use the following command to confirm if the CPU virtualization extension (vmx) is available:

```
$ sudo grep -e 'vmx' /proc/cpuinfo
```

For AMD-based hosts, confirm if the CPU virtualization extension (svm) is available by running the following command:

```
$ sudo grep -e 'svm' /proc/cpuinfo
```

If virtualization is not enabled, you can enable it in the BIOS settings of your machine.

2. Install KVM on Rocky Linux 9 / AlmaLinux 9

Run the following command to install the KVM packages, which are located in the default repository of Rocky Linux 9:

```
$  sudo dnf install qemu-kvm libvirt virt-manager virt-install
```

Recent Posts

How to Setup DHCP Server With Dnsmasq on Debian 12

How to Install SonarQube with Docker Compose on Ubuntu 22.04

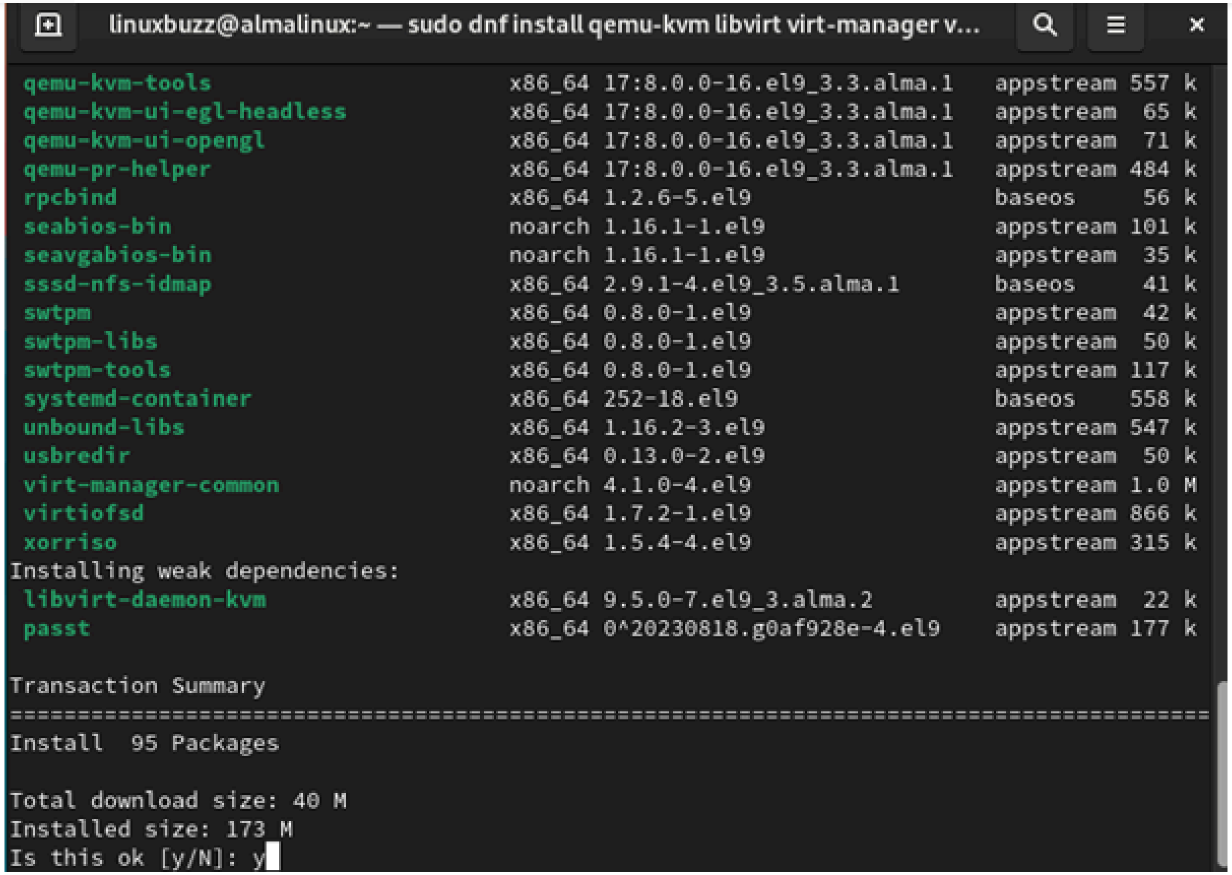
How to Install KVM on Rocky Linux 9 / AlmaLinux 9

How to Install Java on Debian 12

How to Install PostgreSQL 16 on Ubuntu 22.04

14 Useful APT Command Examples in Ubuntu/Debian Linux

How to Setup Bind (DNS Server) on


linuxbuzz@almalinux:~ — sudo dnf install qemu-kvm libvirt virt-manager v...

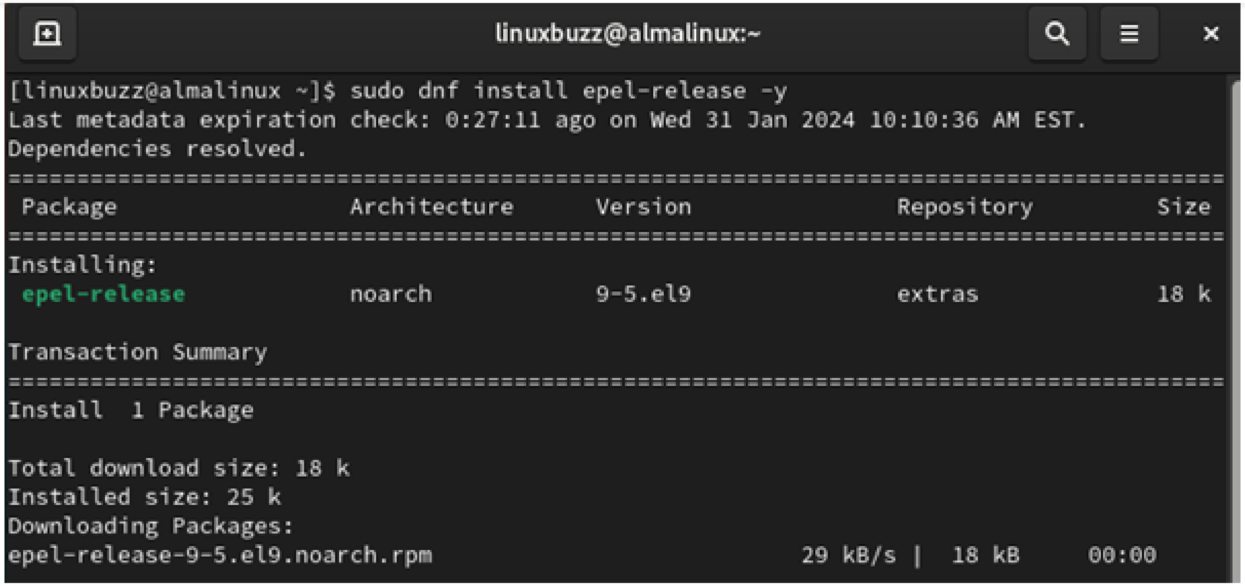
```
 qemu-kvm-tools                     x86_64 17:8.0.0-16.el9_3.3.alma.1   appstream 557 k
 qemu-kvm-ui-egl-headless           x86_64 17:8.0.0-16.el9_3.3.alma.1   appstream  65 k
 qemu-kvm-ui-opengl                 x86_64 17:8.0.0-16.el9_3.3.alma.1   appstream  71 k
 qemu-pr-helper                     x86_64 17:8.0.0-16.el9_3.3.alma.1   appstream 484 k
 rpcbind                            x86_64 1.2.6-5.el9                  baseos     56 k
 seabios-bin                        noarch 1.16.1-1.el9                 appstream 101 k
 seavgabios-bin                     noarch 1.16.1-1.el9                 appstream  35 k
 sssd-nfs-idmap                     x86_64 2.9.1-4.el9_3.5.alma.1       baseos     41 k
 swtpm                              x86_64 0.8.0-1.el9                  appstream  42 k
 swtpm-libs                         x86_64 0.8.0-1.el9                  appstream  50 k
 swtpm-tools                        x86_64 0.8.0-1.el9                  appstream 117 k
 systemd-container                  x86_64 252-18.el9                   baseos    558 k
 unbound-libs                       x86_64 1.16.2-3.el9                 appstream 547 k
 usbredir                           x86_64 0.13.0-2.el9                 appstream  50 k
 virt-manager-common                noarch 4.1.0-4.el9                  appstream 1.0 M
 virtiofsd                          x86_64 1.7.2-1.el9                  appstream 866 k
 xorriso                            x86_64 1.5.4-4.el9                  appstream 315 k
Installing weak dependencies:
 libvirt-daemon-kvm                 x86_64 9.5.0-7.el9_3.alma.2         appstream  22 k
 passt                              x86_64 0^20230818.g0af928e-4.el9    appstream 177 k

Transaction Summary
=========================================================================================
Install  95 Packages

Total download size: 40 M
Installed size: 173 M
Is this ok [y/N]: y
```

Install other KVM management tools as shown:

```
$ sudo dnf install epel-release -y
```

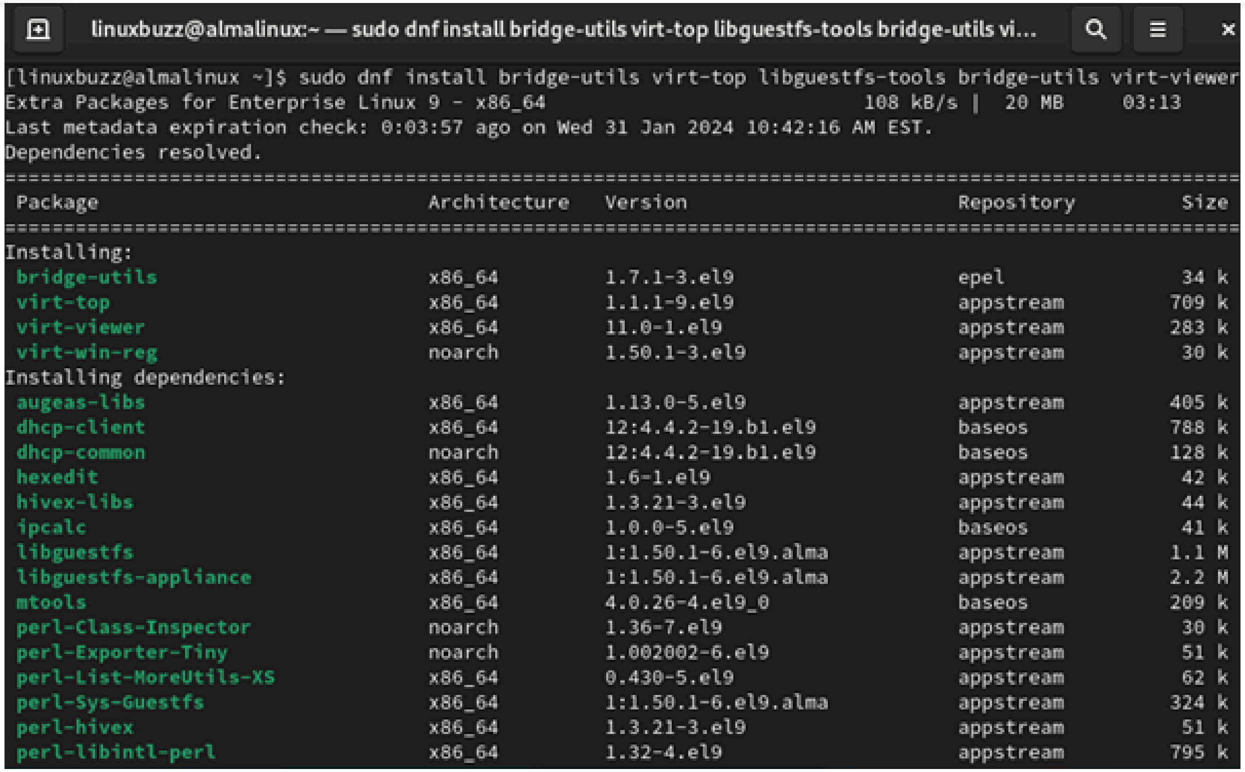
How to Install Kubernetes Cluster on AlmaLinux 9

How to Install Jenkins on Ubuntu 22.04



```
[linuxbuzz@almalinux ~]$ sudo dnf install epel-release -y
Last metadata expiration check: 0:27:11 ago on Wed 31 Jan 2024 10:10:36 AM EST.
Dependencies resolved.
==========================================================================================
 Package                Architecture      Version              Repository          Size
==========================================================================================
Installing:
 epel-release           noarch            9-5.el9              extras              18 k

Transaction Summary
==========================================================================================
Install  1 Package

Total download size: 18 k
Installed size: 25 k
Downloading Packages:
epel-release-9-5.el9.noarch.rpm                        29 kB/s |  18 kB     00:00
```

```
$ sudo dnf -y install bridge-utils virt-top libguestfs-tools bridge-utils virt-vi
```


linuxbuzz@almalinux:~ — sudo dnf install bridge-utils virt-top libguestfs-tools bridge-utils vi...

```
[linuxbuzz@almalinux ~]$ sudo dnf install bridge-utils virt-top libguestfs-tools bridge-utils virt-viewer
Extra Packages for Enterprise Linux 9 - x86_64                          108 kB/s |  20 MB     03:13
Last metadata expiration check: 0:03:57 ago on Wed 31 Jan 2024 10:42:16 AM EST.
Dependencies resolved.
=========================================================================================================
 Package                            Architecture   Version                        Repository        Size
=========================================================================================================
Installing:
 bridge-utils                       x86_64         1.7.1-3.el9                    epel              34 k
 virt-top                           x86_64         1.1.1-9.el9                    appstream        709 k
 virt-viewer                        x86_64         11.0-1.el9                     appstream        283 k
 virt-win-reg                       noarch         1.50.1-3.el9                   appstream         30 k
Installing dependencies:
 augeas-libs                        x86_64         1.13.0-5.el9                   appstream        405 k
 dhcp-client                        x86_64         12:4.4.2-19.b1.el9             baseos           788 k
 dhcp-common                        noarch         12:4.4.2-19.b1.el9             baseos           128 k
 hexedit                            x86_64         1.6-1.el9                      appstream         42 k
 hivex-libs                         x86_64         1.3.21-3.el9                   appstream         44 k
 ipcalc                             x86_64         1.0.0-5.el9                    baseos            41 k
 libguestfs                         x86_64         1:1.50.1-6.el9.alma            appstream        1.1 M
 libguestfs-appliance               x86_64         1:1.50.1-6.el9.alma            appstream        2.2 M
 mtools                             x86_64         4.0.26-4.el9_0                 baseos           209 k
 perl-Class-Inspector               noarch         1.36-7.el9                     appstream         30 k
 perl-Exporter-Tiny                 noarch         1.002002-6.el9                 appstream         51 k
 perl-List-MoreUtils-XS             x86_64         0.430-5.el9                    appstream         62 k
 perl-Sys-Guestfs                   x86_64         1:1.50.1-6.el9.alma            appstream        324 k
 perl-hivex                         x86_64         1.3.21-3.el9                   appstream         51 k
 perl-libintl-perl                  x86_64         1.32-4.el9                     appstream        795 k
```

The following is a brief explanation of the above packages:

- virt-manager provides graphical user interface for managing virtual machines.
- libvirt-client offers CL utility to administer the virtual environment.
- virt-install is the command line tool used to create virtual machines.
- libvirt provides the host-side libraries for interacting with host systems and hypervisors.

After the installation is complete, run the following command check whether KVM module is loaded into the kernel or not.

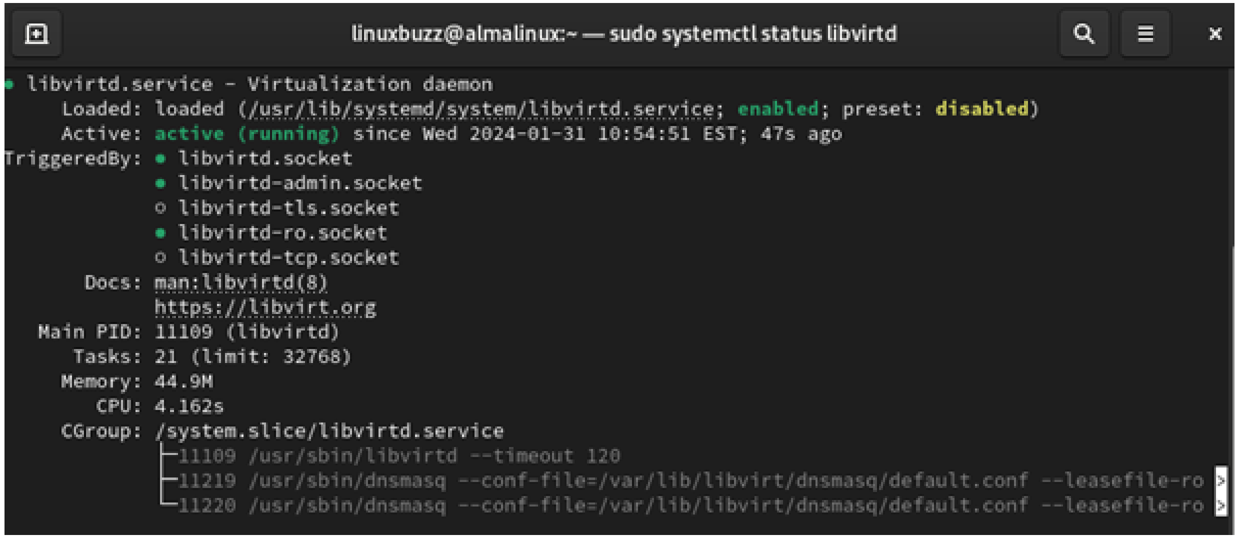
libvirtd is a daemon component that operates on the server side and controls tasks on virtualized guests. It is employed in managing virtualization technologies, including ESXi, KVM, and Xen.

To start and enable the libvirtd daemon, run:

```
$ sudo systemctl start libvirtd
$ sudo systemctl enable libvirtd
```

To check if the libvirtd daemon is running, run the following command:

```
$ sudo systemctl status libvirtd
```



```
● libvirtd.service - Virtualization daemon
     Loaded: loaded (/usr/lib/systemd/system/libvirtd.service; enabled; preset: disabled)
     Active: active (running) since Wed 2024-01-31 10:54:51 EST; 47s ago
TriggeredBy: ● libvirtd.socket
             ● libvirtd-admin.socket
             ○ libvirtd-tls.socket
             ● libvirtd-ro.socket
             ○ libvirtd-tcp.socket
       Docs: man:libvirtd(8)
             https://libvirt.org
   Main PID: 11109 (libvirtd)
      Tasks: 21 (limit: 32768)
     Memory: 44.9M
        CPU: 4.162s
     CGroup: /system.slice/libvirtd.service
             ├─11109 /usr/sbin/libvirtd --timeout 120
             ├─11219 /usr/sbin/dnsmasq --conf-file=/var/lib/libvirt/dnsmasq/default.conf --leasefile-ro >
             └─11220 /usr/sbin/dnsmasq --conf-file=/var/lib/libvirt/dnsmasq/default.conf --leasefile-ro >
```

You must also add your system user to the KVM group to execute virt-install commands

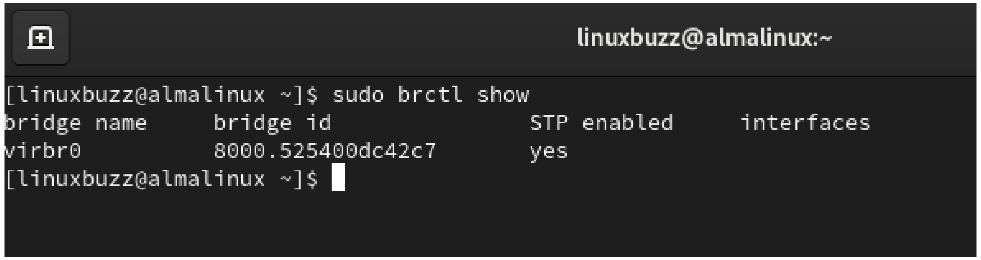
```
$ newgrp libvirt
```

4. Create Network Bridge for KVM Instances

A network bridge with the name virbr0 is automatically created to offer Network Address Translation (NAT). Virtual machines using this bridge lack external connectivity.

Existing bridge networks can be listed using the brctl command:

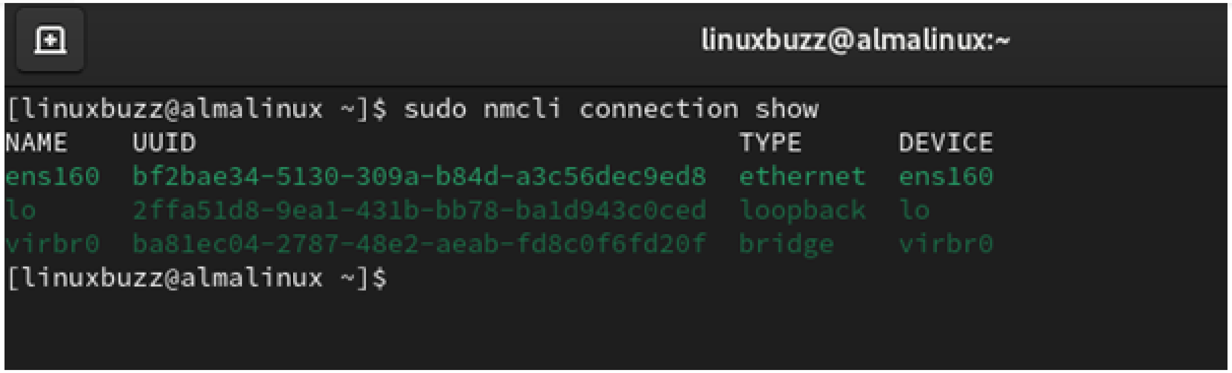
```
$ sudo brctl show
```



```
[linuxbuzz@almalinux ~]$ sudo brctl show
bridge name     bridge id               STP enabled     interfaces
virbr0          8000.525400dc42c7       yes
[linuxbuzz@almalinux ~]$
```

In this section, we'll create a network bridge for external connections using NMCLI. To get started, run the following command to list the network interfaces available on your machine:

```
$ sudo nmcli connection show
```



```
[linuxbuzz@almalinux ~]$ sudo nmcli connection show
NAME     UUID                                  TYPE      DEVICE
ens160   bf2bae34-5130-309a-b84d-a3c56dec9ed8  ethernet  ens160
lo       2ffa51d8-9ea1-431b-bb78-ba1d943c0ced  loopback  lo
virbr0   ba81ec04-2787-48e2-aeab-fd8c0f6fd20f  bridge    virbr0
[linuxbuzz@almalinux ~]$
```

To start creating the bridge, first, delete the existing connection using the following command:

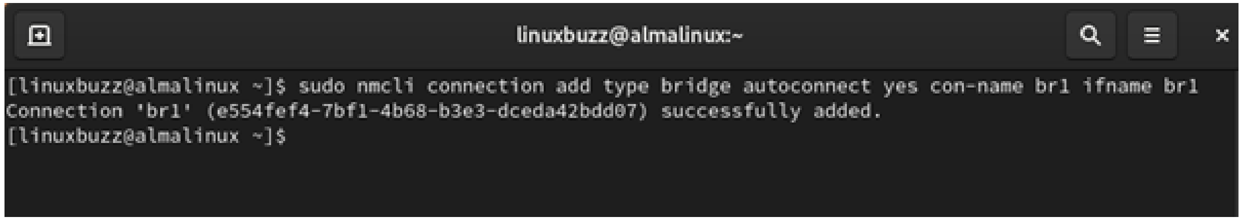
```
$ sudo nmcli connection delete bf2bae34–5130–309a–b84d–a3c56dec9ed8
```

Before moving on, it would be important to have the following information at hand:

- gateway: This is the network's default gateway address, for example, 192.168.16.2
- DNS1 and DNS2: These are the preferred DNS addresses (e.g., 8.8.8.8 and 8.8.4.4).

To create a new bridge, run the following command:

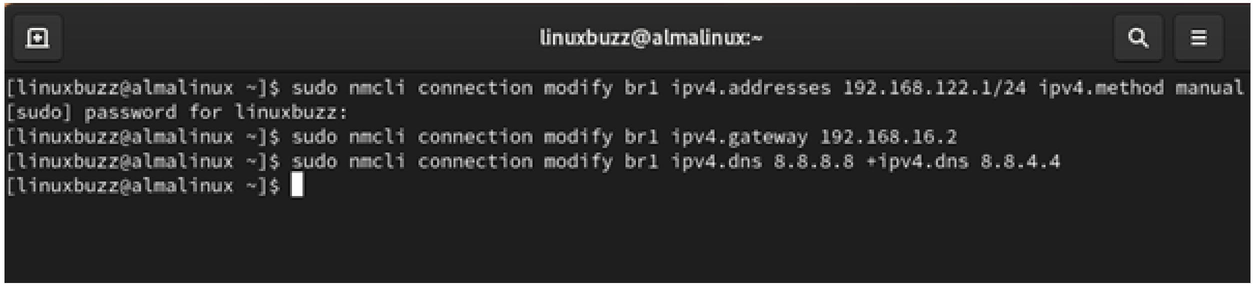
```
$ sudo nmcli connection add type bridge autoconnect yes con-name br1 ifname br1
```



```
[linuxbuzz@almalinux ~]$ sudo nmcli connection add type bridge autoconnect yes con-name br1 ifname br1
Connection 'br1' (e554fef4-7bf1-4b68-b3e3-dceda42bdd07) successfully added.
[linuxbuzz@almalinux ~]$
```

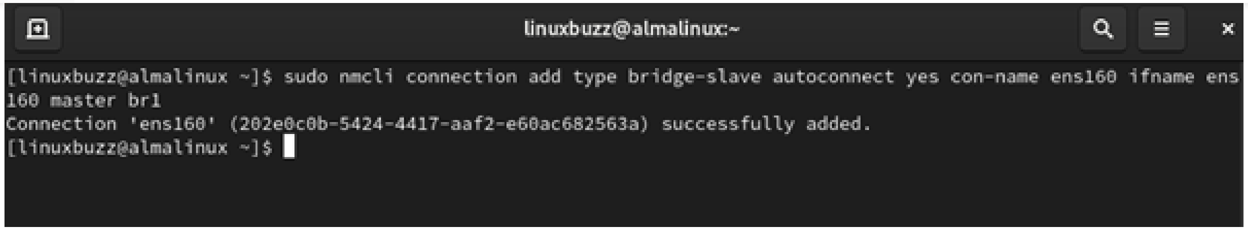
Next, add the IP, gateway, and DNS to the bridge, as shown below:

```
$ sudo nmcli connection modify br1 ipv4.addresses 192.168.16.122.1/24 ipv4.method
$ sudo nmcli connection modify br1 ipv4.gateway 192.168.16.2
$ sudo nmcli connection modify br1 ipv4.dns 8.8.8.8 +ipv4.dns 8.8.4.4
```



```
[linuxbuzz@almalinux ~]$ sudo nmcli connection modify br1 ipv4.addresses 192.168.122.1/24 ipv4.method manual
[sudo] password for linuxbuzz:
[linuxbuzz@almalinux ~]$ sudo nmcli connection modify br1 ipv4.gateway 192.168.16.2
[linuxbuzz@almalinux ~]$ sudo nmcli connection modify br1 ipv4.dns 8.8.8.8 +ipv4.dns 8.8.4.4
[linuxbuzz@almalinux ~]$
```

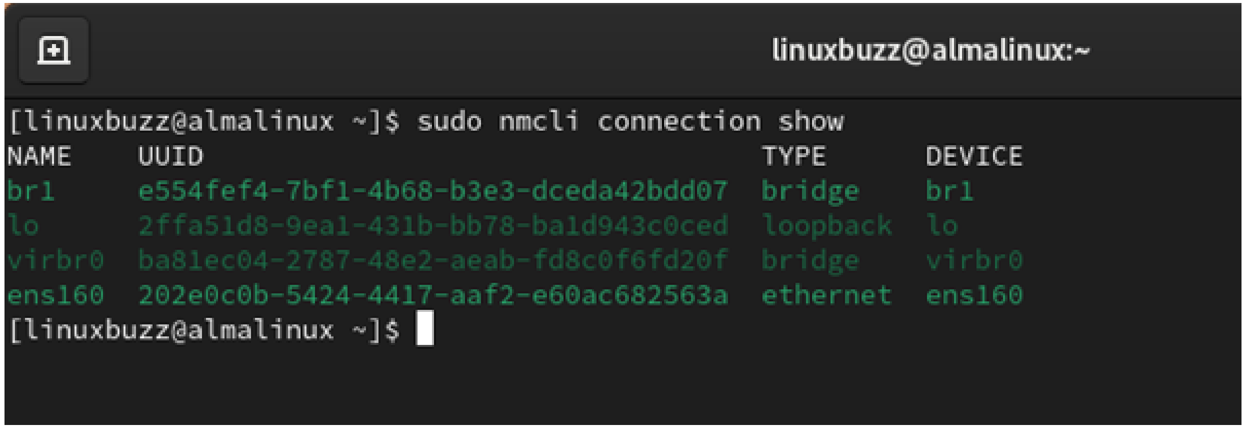
Now, run the following command to add the bridge slave:



```
[linuxbuzz@almalinux ~]$ sudo nmcli connection add type bridge-slave autoconnect yes con-name ens160 ifname ens
160 master br1
Connection 'ens160' (202e0c0b-5424-4417-aaf2-e60ac682563a) successfully added.
[linuxbuzz@almalinux ~]$
```

To verify the bridge creation, run the following command:

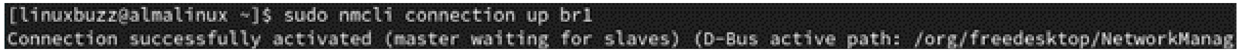
```
$ sudo nmcli connection show
```



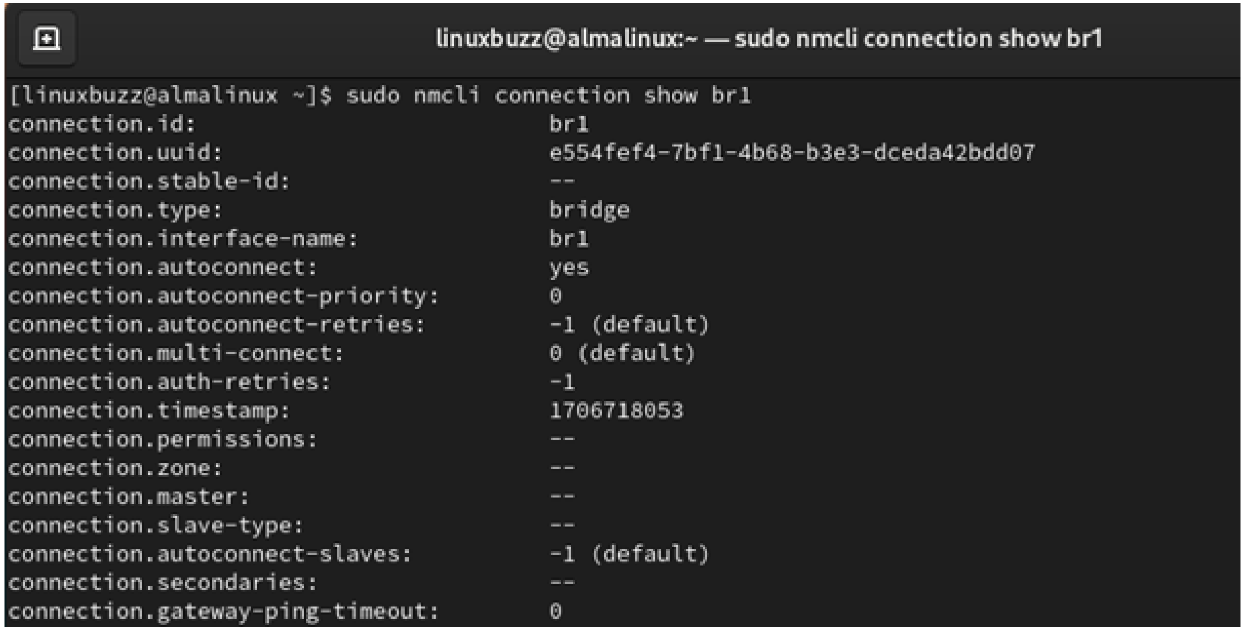
```
[linuxbuzz@almalinux ~]$ sudo nmcli connection show
NAME     UUID                                  TYPE      DEVICE
br1      e554fef4-7bf1-4b68-b3e3-dceda42bdd07  bridge    br1
lo       2ffa51d8-9ea1-431b-bb78-ba1d943c0ced  loopback  lo
virbr0   ba81ec04-2787-48e2-aeab-fd8c0f6fd20f  bridge    virbr0
ens160   202e0c0b-5424-4417-aaf2-e60ac682563a  ethernet  ens160
[linuxbuzz@almalinux ~]$
```

Next, start the network bridge:

```
$ sudo nmcli connection up br1
```



```
[linuxbuzz@almalinux ~]$ sudo nmcli connection up br1
Connection successfully activated (master waiting for slaves) (D-Bus active path: /org/freedesktop/NetworkManag
$ sudo nmcli connection show br1
```

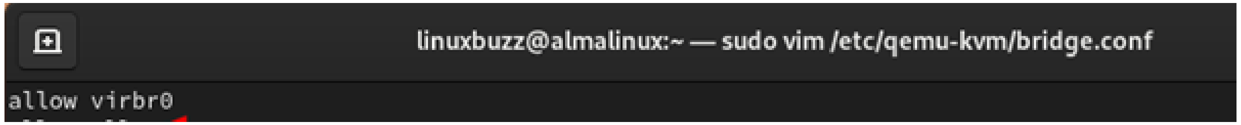


```
[linuxbuzz@almalinux ~]$ sudo nmcli connection show br1
connection.id:                          br1
connection.uuid:                        e554fef4-7bf1-4b68-b3e3-dceda42bdd07
connection.stable-id:                   --
connection.type:                        bridge
connection.interface-name:              br1
connection.autoconnect:                 yes
connection.autoconnect-priority:        0
connection.autoconnect-retries:         -1 (default)
connection.multi-connect:               0 (default)
connection.auth-retries:                -1
connection.timestamp:                   1706718053
connection.permissions:                 --
connection.zone:                        --
connection.master:                      --
connection.slave-type:                  --
connection.autoconnect-slaves:          -1 (default)
connection.secondaries:                 --
connection.gateway-ping-timeout:        0
```

To enable KVM to use this bridge, edit the below file,

```
$ sudo vim /etc/qemu–kvm/bridge.conf
```

Add the line:



```
allow virbr0
```

Then restart libvirtd service

```
$ sudo systemctl restart libvirtd
```

5. Creating Virtual Machine using KVM

Now that KVM is set up and the bridge connection is established, let's create a virtual machine. You need an ISO file to continue with the VM creation.

It's easy to spin a virtual machine via the CLI, especially if you understand the fundamentals of KVM. To get started, set the right ownership of the libvirt directory:

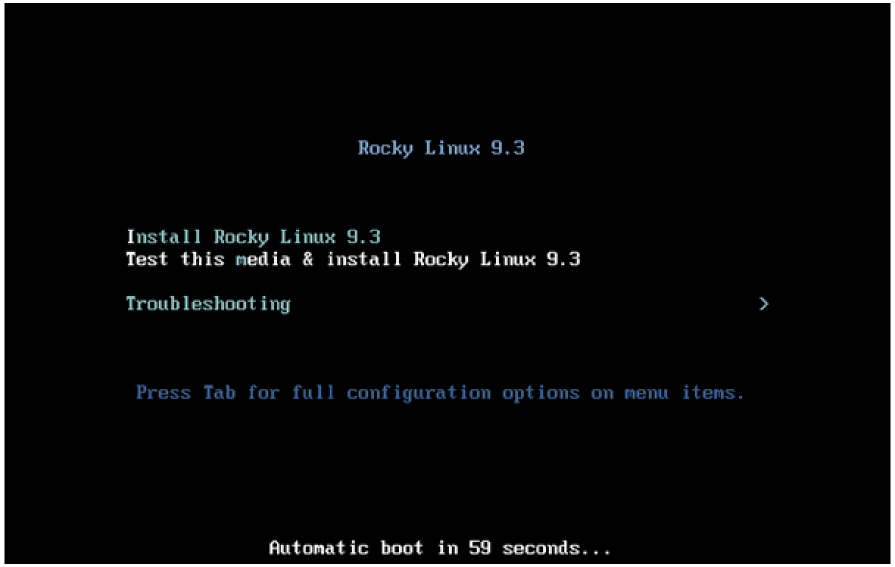
```
$ sudo chown -R $USER:libvirt /var/lib/libvirt/
```

Using the following syntax, we will create a virtual machine on the command line using the Rocky9 Linux image.

```
$ virt-install \
--name Rocky9 \
--ram 2048 \
--vcpus 1 \
--disk path=/var/lib/libvirt/images/rocky-9.img,size=20 \
--os-variant centos-stream9 \
--network bridge=br1,model=virtio \
--graphics vnc,listen=0.0.0.0 \
--console pty,target_type=serial \
--location /home/rocky9/Downloads/Rocky-9.0-x86_64-minimal.iso
```

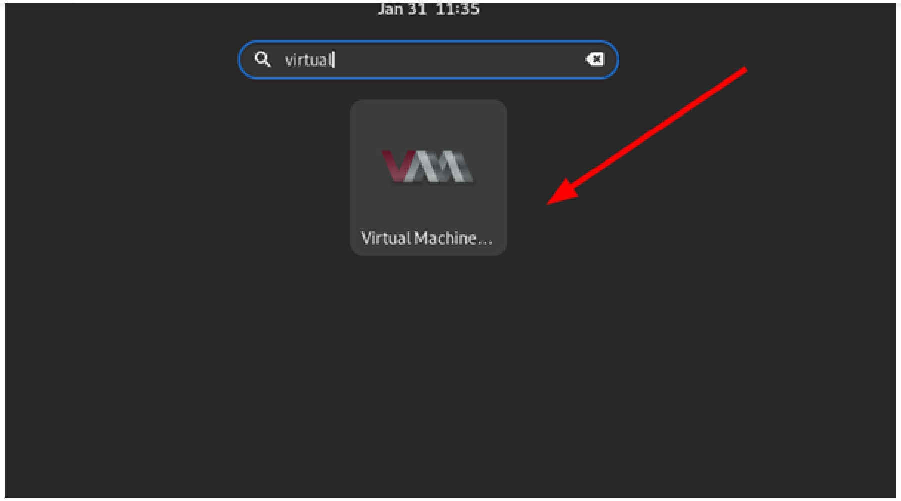
- -ram 2048 is the memory allocated.
- –network bridge=br1 specifies the network bridge to use.
- –graphics vnc,listen=0.0.0.0 shows the VNC listen address.
- –location /home/rocky9/Downloads/Rocky-9.0-x86_64-minimal.iso is the path of your ISO file.

When the above command is executed, VNC will launch and the guest operating system installation will begin, as shown below.

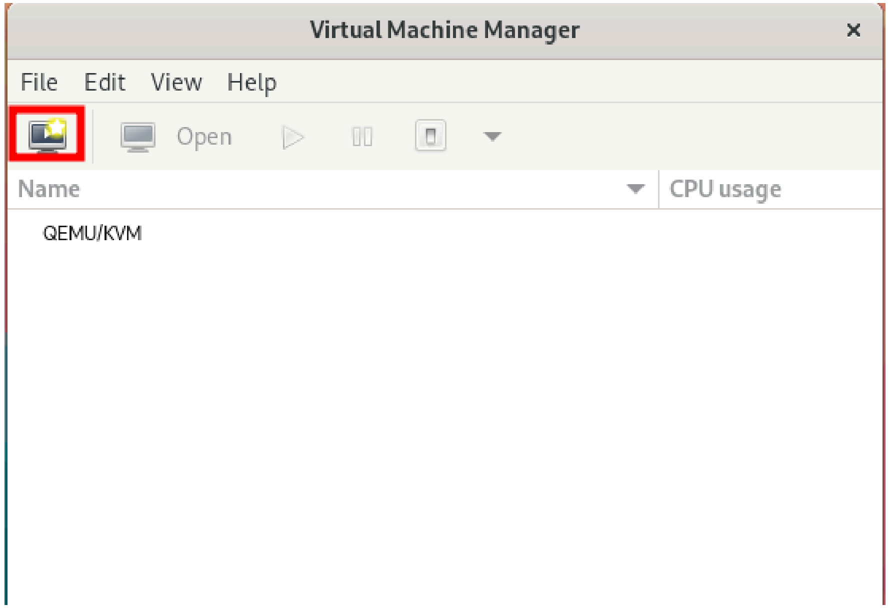


```
                    Rocky Linux 9.3


      Install Rocky Linux 9.3
      Test this media & install Rocky Linux 9.3

      Troubleshooting                                  >


       Press Tab for full configuration options on menu items.




                 Automatic boot in 59 seconds...
```

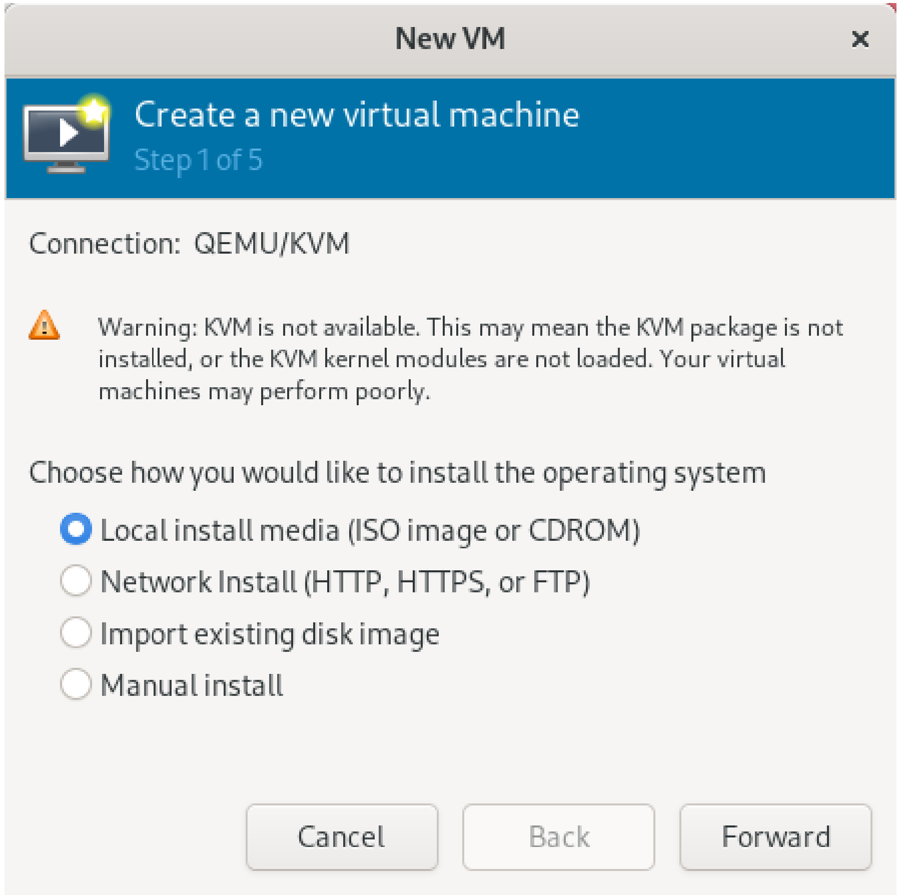
You can also create virtual machines using Virt-Manager GUI. To get started, head over to the application menu and click the icon as shown:



Click on the highlighted icon to create a virtual machine.



Here, select how you would like to install the ISO, choose the default option, and proceed.



Create a new virtual machine

Step 1 of 5

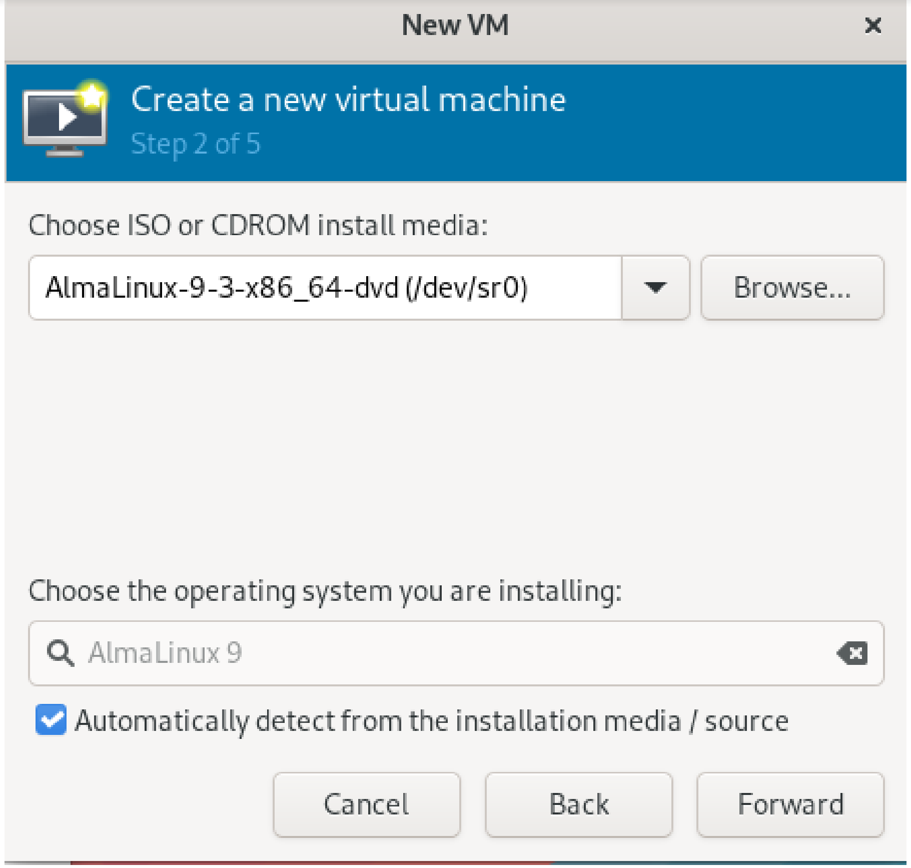
Connection: QEMU/KVM

Warning: KVM is not available. This may mean the KVM package is not installed, or the KVM kernel modules are not loaded. Your virtual machines may perform poorly.

Choose how you would like to install the operating system

● Local install media (ISO image or CDROM)
○ Network Install (HTTP, HTTPS, or FTP)
○ Import existing disk image
○ Manual install

In this window, select the ISO image.



New VM

Create a new virtual machine
Step 2 of 5

Choose ISO or CDROM install media:

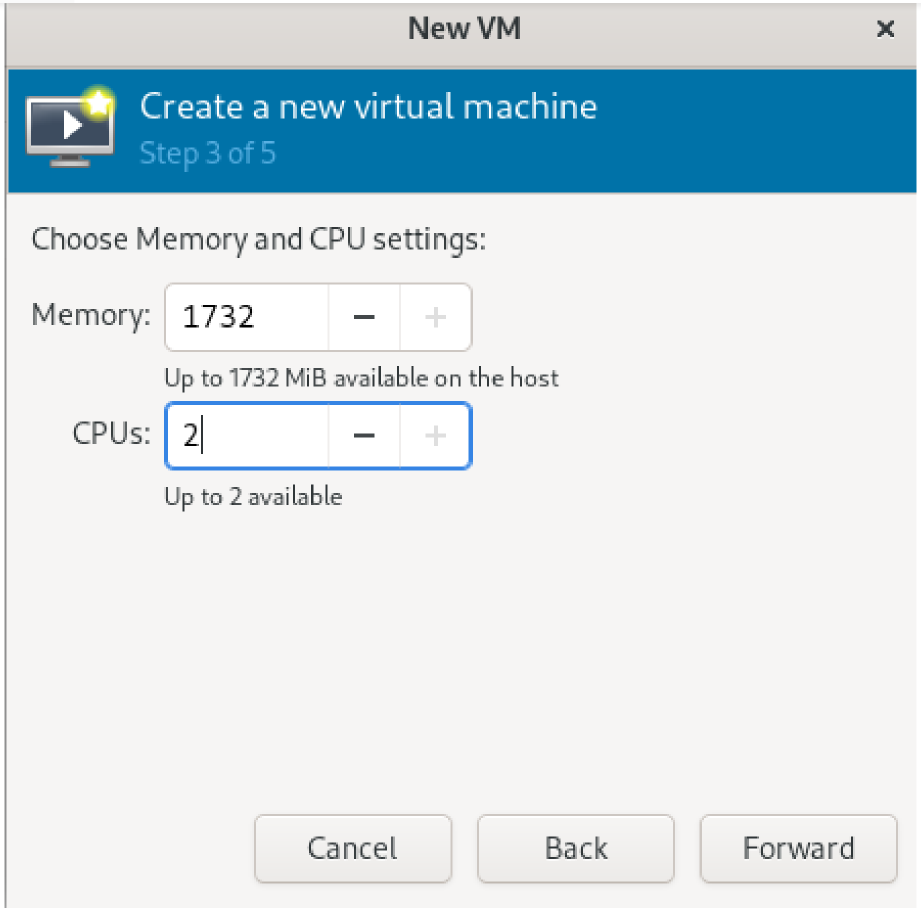
AlmaLinux-9-3-x86_64-dvd (/dev/sr0)

Choose the operating system you are installing:

AlmaLinux 9

☑ Automatically detect from the installation media / source

Set up the virtual machine’s memory and CPU.



Create a new virtual machine
Step 3 of 5

Choose Memory and CPU settings:

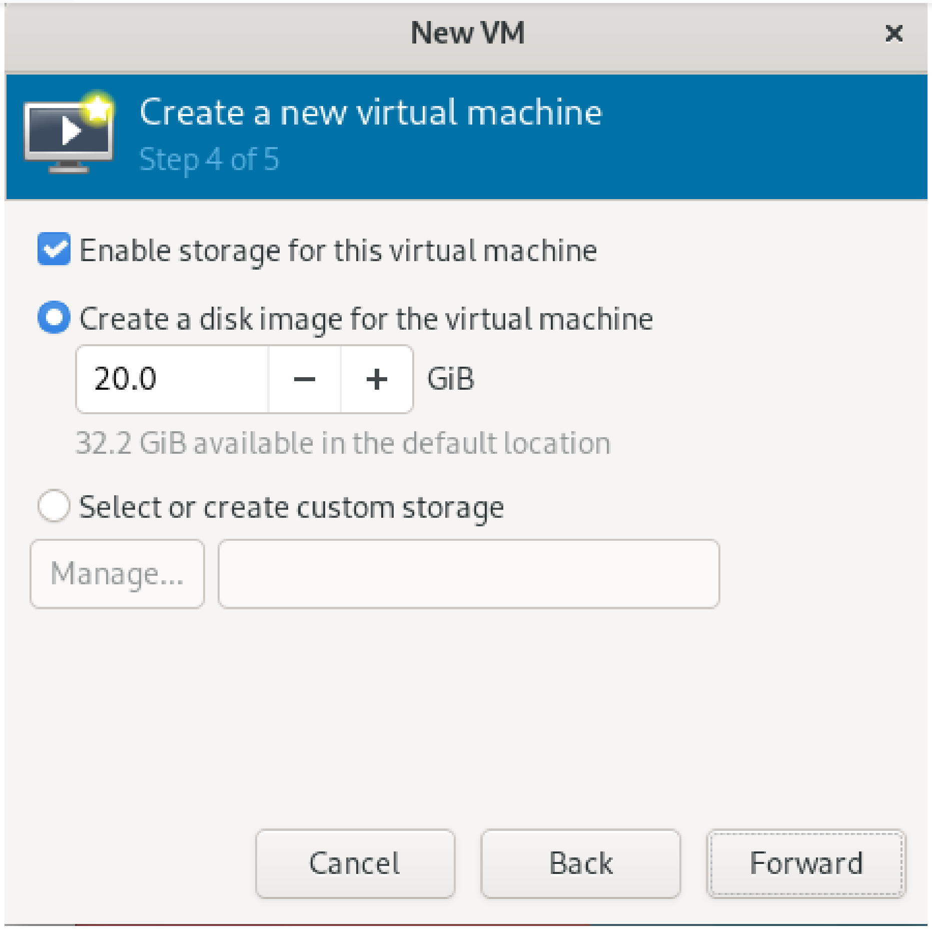
Memory: 1732

Up to 1732 MiB available on the host

CPUs: 2

Up to 2 available

Here, configure the hard disk size.



New VM

Create a new virtual machine

Step 4 of 5

☑ Enable storage for this virtual machine

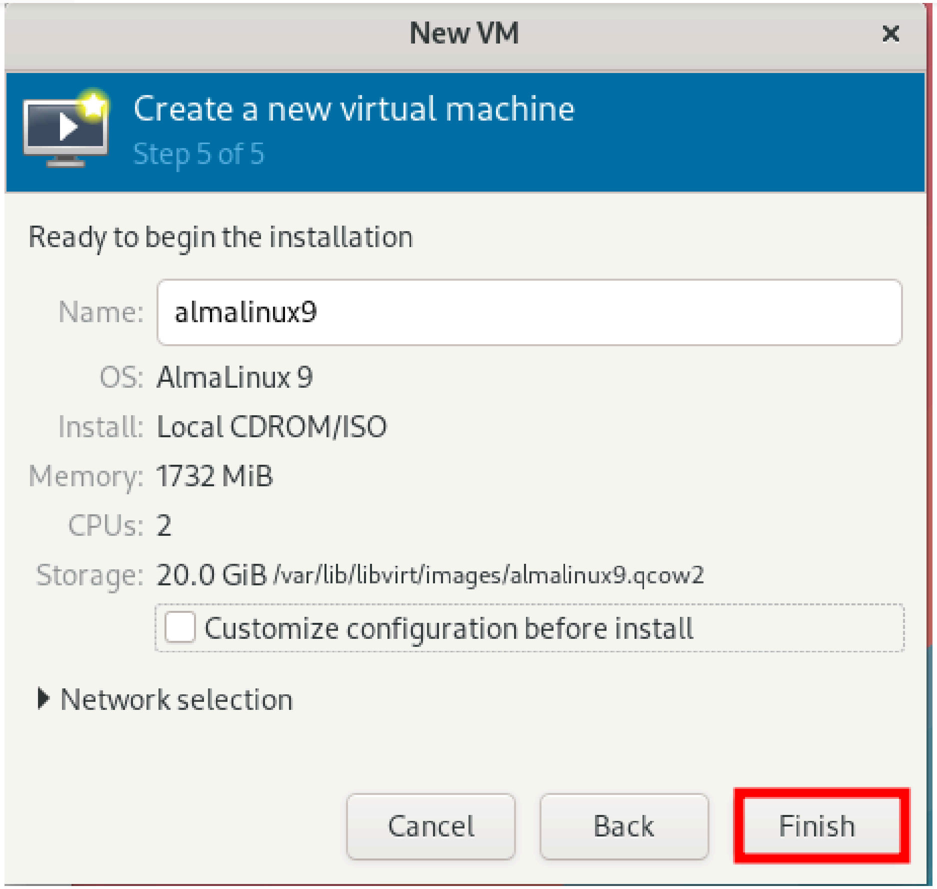
● Create a disk image for the virtual machine

20.0 GiB

32.2 GiB available in the default location

○ Select or create custom storage

Now, click the finish button for the installation to begin.



Create a new virtual machine

Step 5 of 5

Ready to begin the installation

Name: almalinux9

OS: AlmaLinux 9

Install: Local CDROM/ISO

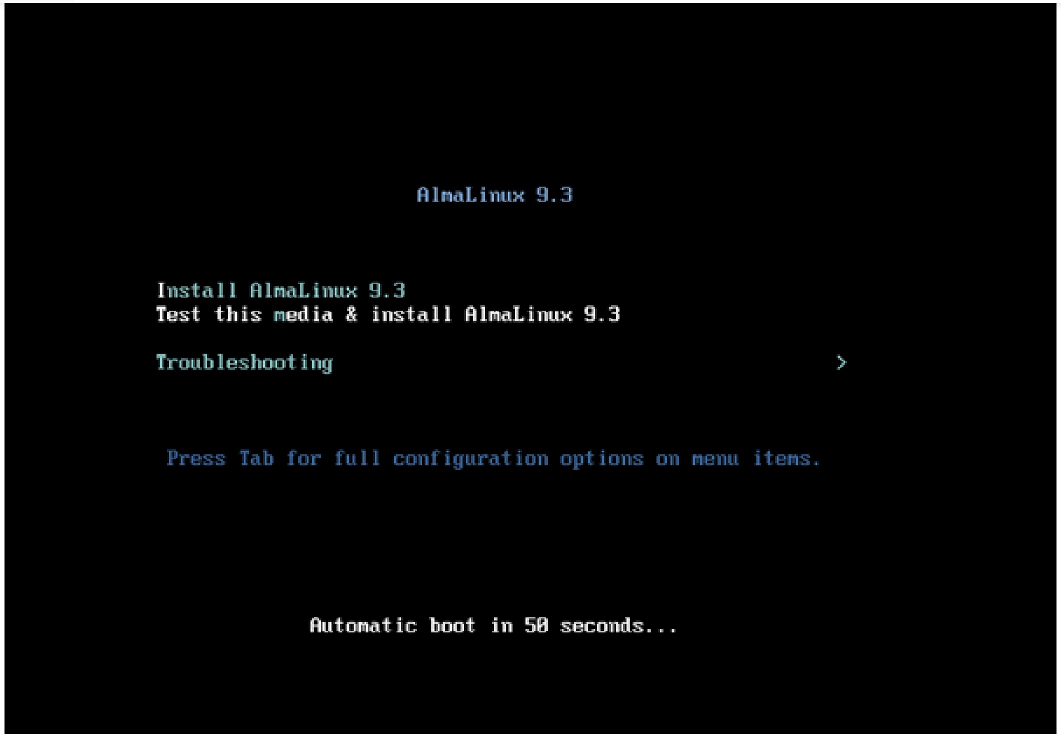
Memory: 1732 MiB

CPUs: 2

Storage: 20.0 GiB /var/lib/libvirt/images/almalinux9.qcow2

□ Customize configuration before install

▸ Network selection



```
AlmaLinux 9.3

Install AlmaLinux 9.3
Test this media & install AlmaLinux 9.3

Troubleshooting                                    >

Press Tab for full configuration options on menu items.

Automatic boot in 50 seconds...
```

Conclusion

That’s it! We have shown you how to install KVM on Rocky Linux 9 or AlmaLinux 9. We hope you find this tutorial useful and informative. Feel free to post your queries and feedback in below comments section.

Leave a Comment

Name *

Email *

Website

Please enter an answer in digits:

four × 1 =

Post Comment