shamil / **mount_qcow2.md**

Last active 10 hours ago

How to mount a qcow2 disk image

`mount_qcow2.md`

How to mount a qcow2 disk image

This is a quick guide to mounting a qcow2 disk images on your host server. This is useful to reset passwords, edit files, or recover something without the virtual machine running.

Step 1 - Enable NBD on the Host

```
modprobe nbd max_part=8
```

Step 2 - Connect the QCOW2 as network block device

```
qemu-nbd --connect=/dev/nbd0 /var/lib/vz/images/100/vm-100-disk-1.qcow2
```

Step 3 - Find The Virtual Machine Partitions

```
fdisk /dev/nbd0 -l
```

Step 4 - Mount the partition from the VM

```
mount /dev/nbd0p1 /mnt/somepoint/
```

Step 5 - After you done, unmount and disconnect

```
umount /mnt/somepoint/
qemu-nbd --disconnect /dev/nbd0
rmmod nbd
```

poma commented on Jan 27, 2018 • edited

In my case partition didn't receive its own file

```
root@master# fdisk /dev/nbd0 -l
Disk /dev/nbd0: 501 GiB, 537944653824 bytes, 1050673152 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: dos
Disk identifier: 0x000198dd

Device      Boot Start        End    Sectors  Size Id Type
/dev/nbd0p1         63 1050673151 1050673089  501G 83 Linux

root@master# mount /dev/nbd0p1 /mnt/web/
mount: special device /dev/nbd0p1 does not exist
```

Fixed by running `partx -a /dev/nbd0`

joshenders commented on Feb 6, 2018 • edited

Thanks @**poma**. I had to run `partx -a /dev/nbd0` to create the `/dev/nbd0p*` device nodes before they could be mounted as well on Ubuntu 14.04

GraysonPeddie commented on Jan 1, 2019

I found the step-by-step instructions in Google search engine and yours is very useful. Thanks.

I have shared your step-by-step instructions in Twitter.
https://twitter.com/GraysonPeddie/status/1079875947111821313

ykfq commented on Jan 11, 2019 • edited

Error:

> modprobe: FATAL: Module nbd not found.

```
# modprobe nbd max_part=8
modprobe: FATAL: Module nbd not found.

# cat /etc/redhat-release
CentOS Linux release 7.3.1611 (Core)

# uname -a
Linux controller50 3.10.0-514.el7.x86_64 #1 SMP Tue Nov 22 16:42:41 UTC 2016 x86_64 x86_64 x86_64 GNU/Linux

# gcc -v
Using built-in specs.
COLLECT_GCC=gcc
COLLECT_LTO_WRAPPER=/usr/libexec/gcc/x86_64-redhat-linux/4.8.5/lto-wrapper
Target: x86_64-redhat-linux
...
gcc version 4.8.5 20150623 (Red Hat 4.8.5-11) (GCC)
```

SwetaleenaDash commented on Jan 17, 2019

command:
root@GTAPC:~# qemu-nbd --connect /dev/nbd1 /root/Automation/Gold/Working/QCOW/QCOW_VM1.qcow
Error:
Failed to open /dev/nbd1: No such file or directory
Disconnect client, due to: End of file

what doe this error mean?
(/root/Automation/Gold/Working/QCOW/QCOW_VM1.qcow is path of QCOW file)

mluppov commented on Jan 19, 2019 • edited

> modprobe nbd max_part=8
> modprobe: FATAL: Module nbd not found.

Since modprobe is a tool for loading kernel modules, the support for network block device aka nbd in your kernel was not compiled as a module. Try to just skip this part and go to step 2. In case it will fail too, you don't have nbd support at all. In this case you need to include it.

mluppov commented on Jan 19, 2019

> Error:
> Failed to open /dev/nbd1: No such file or directory

Most likely you either didn't load kernel module first or it failed to load and you did not pay attention to the error message. It may also be you don't have nbd support in your kernel at all.

SwetaleenaDash commented on Feb 28, 2019 • edited

while doing step 4, i am getting the issue "mount: special device /dev/nbd1p2 does not exist". why am i getting this error?
after running "partx -a /dev/nbd0" also i am getting the error "partx: /dev/nbd1: failed to read partition table"

MuralidharB commented on Oct 14, 2019 • edited

nbd module is not shipped with Fedora based distributions including centos/fedora/rhel. RedHat decided against shipping nbd kernel module due to security reasons. nbd module is only available on debian distributions.

Otherwise qemu-nbd is a nice tool to mount qcow2 images. The only other option is to use guestfish to mount and access individual files from qcow2 images.

llegolas commented on Jan 7, 2020

> nbd module is not shipped with Fedora based distributions including centos/fedora/rhel. RedHat decided against shipping nbd kernel module due to security reasons. nbd module is only available on debian distributions.
>
> Otherwise qemu-nbd is a nice tool to mount qcow2 images. The only other option is to use guestfish to mount and access individual files from qcow2 images.

Not entirely correct.

```
$ lsb_release -a
LSB Version:    :core-4.1-amd64:core-4.1-noarch
Distributor ID: Fedora
Description:    Fedora release 31 (Thirty One)
Release:        31
Codename:       ThirtyOne
$ rpm -qf /usr/lib/modules/$(uname -r)/kernel/drivers/block/nbd.ko.xz
kernel-core-5.3.16-300.fc31.x86_64
$ sudo modprobe nbd max_part=8
$ lsmod | grep nbd
nbd                    49152  0
```

MuralidharB commented on Jan 7, 2020

I stand corrected then. I only tested with CentOS and Red Hat and I assumed Fedora falls into the same bucket. If nbd driver is available, then it is a better option than guest fish based mount.

N0NB commented on Feb 26, 2020

Works perfectly on Debian 10.

faywong commented on Mar 10, 2020

It doesn't work in my situation:

```
NTFS signature is missing.
Failed to mount '/dev/nbd0p2': invalid arguments
The device '/dev/nbd0p2' doesn't seem to have a valid NTFS.
Maybe the wrong device is used? Or the whole disk instead of a
partition (e.g. /dev/sda, not /dev/sda1)? Or the other way around?
```

faywong commented on Mar 10, 2020

and filesystem type:

```
sudo file -s /dev/nbd0p2
/dev/nbd0p2: data
```

faywong commented on Mar 10, 2020

```
lsmod |grep -i nbd
nbd                    45056  2
```

fluttr commented on Jul 20, 2020 • edited

Thanks for the tip! There is another simple way to do this though:
`guestmount -a path_to_image.qcow2 -i --ro /mount_point # safe, read only`
`guestmount -a path_to_image.qcow2 -i /mount_point # use only on not running vm image`
guestmount utility can be found in libguestfs-tools package (on Debian and RHEL).

willzhang commented on Jul 31, 2020 • edited

kernel must be 4.10.xx?

```
[root@rave-pony-1 ~]# cat /etc/redhat-release
CentOS Linux release 7.8.2003 (Core)

[root@rave-pony-1 ~]# uname -sr
Linux 4.10.4-1.el7.elrepo.x86_64

[root@rave-pony-1 ~]# modprobe nbd max_part=8
[root@rave-pony-1 ~]# ls /dev/nbd*
/dev/nbd0  /dev/nbd10  /dev/nbd12  /dev/nbd14  /dev/nbd2  /dev/nbd4  /dev/nbd6  /dev/nbd8
/dev/nbd1  /dev/nbd11  /dev/nbd13  /dev/nbd15  /dev/nbd3  /dev/nbd5  /dev/nbd7  /dev/nbd9
```

3.10.x kernel

```
[root@virt-tool ~]# cat /etc/redhat-release
CentOS Linux release 7.8.2003 (Core)

[root@virt-tool ~]# uname -sr
Linux 3.10.0-1127.el7.x86_64

[root@virt-tool ~]#  modprobe nbd max_part=16
modprobe: FATAL: Module nbd not found.
[root@virt-tool ~]#  ls /dev/nbd*
ls: cannot access /dev/nbd*: No such file or directory
```

MaxMatti commented on Jan 18, 2021

> you kill my arch Linux angryangryangryangryangryangryangryangryangryangryangryangryangryangryangry

If you don't know how to mount a filesystem then you shouldn't use archlinux. Have a look at the documentation for the `mount` command to learn how to fix your problem.

0xcpu commented on Jan 26, 2021 • edited

I have some issues with `step 4`.
`step 3` output:

```
Disk /dev/nbd0: 8 GiB, 8589934592 bytes, 16777216 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: gpt
Disk identifier: 0CBEE71E-00C3-470C-B2C4-74D20202393E

Device        Start      End  Sectors Size Type
/dev/nbd0p1    2048     4095     2048   1M BIOS boot
/dev/nbd0p2    4096  2101247  2097152   1G Linux filesystem
/dev/nbd0p3 2101248 16775167 14673920   7G Linux filesystem
```

Then `mount -o ro /dev/nbd0p1 /mnt/fs_mnt/`

```
wrong fs type, bad option, bad superblock on /dev/nbd0p1, missing codepage or helper program, or other error.
```

I guess, the issue is because it's a GPT disk.

Any idea how this can be solved? (Tried fixing the drive with `fsck`, but it didn't help.)

MaxMatti commented on Jan 27, 2021 • edited

@0xcpu:

See this link

> As far as I know the "BIOS Boot Partition" does not contain any filesystem but only the Grub image. It is the space you need that would otherwise be put somewhere between the MBR and the first partition.

Seems like you wouldn't wan to "mount" that but rather just mount the other two partitions.

SKGEzhil commented on Jan 31, 2021

I successfully mounted but I can't able copy and paste files

chrisgraf commented on Apr 7, 2021

thanks! found that via google!

jussihirvi commented on May 30, 2021

I had the same problem as **@poma**. `fdisk` listed the partitions but there were no corresponding device files. `partx` did not help. Run as `partx -v -a /dev/nbd0` it complained about "invalid argument" and did not create anything.

`kpartx` helped. First I checked with `kpartx -l /dev/nbd0`, then actually created the device files with `kpartx -a /dev/nbd0`. The device files appeared in `/dev/mapper`.

jussihirvi commented on May 30, 2021

I had the same problem as **@poma**. `fdisk` listed the partitions but there were no corresponding device files. `partx` did not help. Run as `partx -v -a /dev/nbd0` it complained about "invalid argument" and did not create anything.

`kpartx` helped. First I checked with `kpartx -l /dev/nbd0`, then actually created the device files with `kpartx -a /dev/nbd0`. The device files appeared in `/dev/mapper`.

AggamR commented on Jun 30, 2021

thanks! very useful!

cindrmon commented on Jul 25, 2021

gives me this error, since i made my qcow2 an LVM

```
“mount: unknown filesystem type LVM2_member”
```

Is there any other way for me to mount this as an LVM?

ololobster commented on Dec 13, 2021 • edited

> Is there any other way for me to mount this as an LVM?

Hello from December 2021 :)

1. Run `sudo pvs` to get a volume group (column VG) for your device.
2. Run `sudo lvdisplay ⟨your VG⟩` to get a LV Path.
3. Run `sudo vgchange -a y` to activate all groups.
4. Now you can mount it: `sudo mount ⟨your LV Path⟩ /mnt/somepoint`