

Network Block Devices qemu-nbd kvm-nbd nbd-server nbd-client

(Link zu dieser Seite als [[QEMU-KVM-Buch/ Speichermedien/ Network Block Devices]])

<<<|###| >>> | English

Network Block Devices

Network Block Devices (<http://nbd.sourceforge.net>) sind virtuelle Speichermedien, die von einem NBD-Server über TCP/IP bereitgestellt werden. Als Network Block Devices (NBD) können Festplatten, Festplattenpartition, CDs/DVDs, Disketten oder Image-Dateien auf dem NBD-Server dienen. Andere Rechner können sich über eine TCP/IP-Verbindung mit dem NBD-Server verbinden und die Network Block Devices wie eigene lokale Festplatten benutzen. Die Zugriffsgeschwindigkeit ist, bedingt durch die Übertragungsrate im Netzwerk, nicht hoch. Ab Kernel 2.6.26 unterstützt Linux die Network Block Devices. Das Kernel-Modul *nbd* legt dazu die Devices */dev/nb** an. Unter Linux beinhaltet das Paket *qemu-kvm* die speziellen NBD-Server *qemu-nbd* beziehungsweise *kvm-nbd*. Mit diesen Tools werden Network Block Devices zur Verfügung gestellt. Mit der Option *-k* wird ein Unix-Socket verwendet. Die Option *-t* bewirkt, dass das Programm nicht beim Abbruch der Verbindung beendet wird. Hier als Beispiel der Aufruf von *qemu-nbd*. Bei *kvm-nbd* werden die gleichen Optionen angewendet.

```
Host ~$ qemu-nbd -t -k /var/lock/qemu-nbd Platte.img
```

Auf einer zweiten Konsole des Host-Systems wird ein Gast-System von diesem Network Block Device gestartet.

```
Host ~$ qemu-system-x86_64 -hda nbd:unix:/var/lock/qemu-nbd
```

Wird ein Network Block Device über TCP/IP exportiert, ist der Port anzugeben. Wird der Default-Port 1024 verwendet, kann diese Angabe entfallen. Im folgenden Beispiel wird die Image-Datei *Platte.img* per TCP/IP über das Port *1025* als Network Block Device zur Verfügung gestellt.

```
Host ~$ qemu-nbd -t -p 1025 Platte.img
```

Auf einer zweiten Konsole des Host-Systems kann eine Instanz von diesem Image booten.

```
Host ~$ qemu-system-x86_64 -hda nbd:localhost:1025
```

Soll von einem anderen Rechner auf dieses Network Block Device zugegriffen werden, so ist der Name oder die IP-Adresse des NBD-Servers anzugeben.

```
Host2 ~$ qemu -hda nbd:my_nbd_server:1025
```

Die Option *--shared=num* ermöglicht, dass mehrere Gast-Systeme auf ein Network Block Device zugreifen können. Die maximale Anzahl definiert der Wert *num* (Default = 1).

```
Host1 ~$ qemu-nbd -t -p 1025 --share=2 Platte.img
```

Auf einem anderen Rechner kann eine Instanz mit diesem Network Block Device booten.

```
Host2 ~$ qemu -hda nbd:my_nbd_server:1025
```

Auf einem weiteren Rechner kann eine zweite Instanz gestartet werden.

```
Host3 ~$ qemu -hda nbd:my_nbd_server:1025
```

Mit der Option *-s* oder *--snapshot* werden die Änderungen nicht auf das Image, sondern in temporäre Dateien geschrieben. Für das Gast-System ist scheinbar das Image beschreibbar. Diese Änderungen gehen beim Herunterfahren der Instanz verloren. Diese Option verlangsamt den Zugriff.

```
Host ~$ qemu-nbd -s -t -p 1025 Platte.img
```

Ein kompletter Schreibschutz aktiviert die Option *-r* (*--read-only*). Viele Betriebssysteme können nicht von einem schreibgeschützten Medium booten. Deshalb ist diese Option eher für Live-CDs oder zusätzliche

Speichermedien interessant. Im folgenden Beispiel werden zwei Network Block Devices gleichzeitig auf unterschiedlichen Ports zur Verfügung gestellt. Das Image *Daten.img* ist schreibgeschützt.

```
Host1 ~$ qemu-nbd -s -t -p 1025 Platte
Host1 ~$ qemu-nbd -r -t -p 1026 Daten.img
```

Beide Network Block Devices können von einer Instanz eingebunden werden.

```
Host2 ~$ qemu -hda nbd:my_nbd_server:1025 \
           -hdb nbd:my_nbd_server:1026
```

Neben Images können auch CDs, DVDs und Disketten als Network Block Device exportiert werden. So können unterschiedliche Installationsmedien auf einen zentralen Server bereitgestellt werden.

```
Host1 ~$ qemu-nbd -t -p 1025 /dev/cdrom
```

Dieses Network Block Device wird von einer Instanz auf einen anderen Rechner eingebunden.

```
Host2 ~$ qemu -cdrom nbd:my_nbd_server:1025
```

Mit der Option `--connect=DEV` kann das Image direkt einem konkreten Network Block Device zugewiesen werden. Der Parameter *DEV* kennzeichnet das Device.

```
Host ~$ qemu-nbd --connect=/dev/nbd0 Platte.img
```

Es ist möglich dieses Device zu partitionieren oder zu mounten.

```
Host ~$ sudo fdisk -l /dev/nbd0
Host ~$ sudo mount /dev/nbd0p1 /mnt
```

Alle Optionen von *qemu-nbd* zeigt die Option `--help` an.

```
Host1 ~$ qemu-nbd -help
Usage: qemu-nbd [OPTIONS] FILE
QEMU Disk Network Block Device Server
-p, --port=PORT          port to listen on (default '1024')
-o, --offset=OFFSET     offset into the image
-b, --bind=IFACE        interface to bind to (default '0.0.0.0')
-k, --socket=PATH       path to the unix socket
                        (default '/var/lock/qemu-nbd-DEVICE')
-r, --read-only         export read-only
-P, --partition=NUM     only expose partition NUM
-s, --snapshot          use snapshot file
-n, --nocache           disable host cache
-c, --connect=DEV       connect FILE to the local NBD device DEV
-d, --disconnect        disconnect the specified device
-e, --shared=NUM        device can be shared by NUM clients
-t, --persistent        don't exit on the last connection
-v, --verbose           display extra debugging information
-h, --help              display this help and exit
-V, --version           output version information and exit
```

Als schnellere Alternative zu *qemu-nbd* kann unter Linux das Paket *nbd-server* oder für den Cluster-Einsatz das Paket *gnbd-server* verwendet werden. *nbd-server* bietet einen besseren Zugriffsschutz als *qemu-nbd*.

```
Host ~$ sudo apt-get install nbd-server
```

Um ein Image als Network Block Device zur Verfügung zu stellen, ist der Port und das Image mit dessen Pfad anzugeben. Empfehlenswert ist, mit der Option `-a` ein Timeout in Sekunden anzugeben. Damit verhindert man, dass unnötige Prozesse bei längerer Inaktivität im Speicher verbleiben.

```
Host1 ~$ nbd-server 1025 /Pfad/Platte.img -a 100
```

Von einem zweiten Rechner wird ein Gast-System von diesem Network Block Device gestartet.

```
Host2 ~$ qemu -hda nbd:my_nbd_server:1025
```

Nach Beenden der Instanz ist zu kontrollieren, ob noch der Prozess des NBD-Server aktiv ist.

```
Host1 ~$ ps aux | grep nbd-serve[r]
```

Nicht mehr benötigte Prozesse sind zu beenden.

```
Host1 ~$ killall nbd-server
```

Ein kompletter Schreibschutz für das Image aktiviert die Option `-r`.

```
Host ~$ nbd-server 1025 /Pfad/Platte.img -r -a 100
```

Mit der Option `-c` (Copy on write) werden die Änderungen nicht auf das Image geschrieben, sondern in temporäre Dateien (`*.diff`). Für das Gast-System ist scheinbar das Image beschreibbar. Diese Änderungen gehen beim Herunterfahren der Instanz verloren. Diese Option verlangsamt den Zugriff.

```
Host ~$ nbd-server 1025 /Pfad/Platte.img -c -a 100
```

Alle Optionen zeigt die Man-Page an.

```
Host ~$ man nbd-server
```

Für den Zugriff auf Network Block Devices kann unter Linux das Paket `nbd-client` beziehungsweise `gnbd-client` verwendet werden. Diese NBD-Clients unterstützen nicht das Booten von einem Network Block Device.

```
Host ~$ sudo apt-get install nbd-client
```

Wurde ein Network Block Device mit `qemu-nbd` oder `nbd-server` zur Verfügung gestellt, kann mit dem `nbd-client` darauf zugegriffen werden. Dazu ist die IP-Adresse oder der Host-Name des Servers, das Port und das lokale Device anzugeben.

```
Host ~$ sudo nbd-client my_nbd_server 1025 /dev/nbd0
```

Ein installiertes Gast-System kann mit QEMU gestartet werden.

```
Host ~$ sudo chmod 777 /dev/nbd0
```

```
Host ~$ qemu-system-x86_64 -hda /dev/nbd0
```

Alle Optionen zeigt die Man-Page an.

```
Host ~$ man nbd-client
```

```
<<<|###|>>>
```

Von „http://qemu-buch.de/de/index.php?title=QEMU-KVM-Buch/_Speichermedien/_Network_Block_Devices“

Diese Seite wurde bisher 26.888 mal abgerufen. Diese Seite wurde zuletzt am 23. November 2011 um 18:17 Uhr geändert. Inhalt ist verfügbar unter der GNU Free Documentation License 1.2.