

**FreeBSD 10.3 mit ZFS und bhyve (BSD-Hypervisor)
Virtualisierung von Oracle Linux 7.2
Wolfgang Schraml
18.05.2016**

Hardware:

CPU mit Virtualisierungs-Technik (neuere Intel-CPU's und AMD-CPU's)
Hinweise: <https://www.freebsd.org/doc/handbook/virtualization-host-bhyve.html>

Basis-Installation von FreeBSD (aktuell: 10.3) mit zfs und 8 MB RAM oder mehr

Für die VM mit bhyve ein **eigenes Dataset** (Dateisystem) anlegen, hier z. B. zroot/lug:

```
zfs list  
zfs create -o compress=lz4 zroot/lug  
zfs set atime=off zroot/lug
```

Das Dateisystem kann so später mit dem Befehl **zfs snapshot** getrennt gesichert werden;
bhyve selbst hat noch keine snapshot-Funktion.

Vorbereitungen für bhyve:

Kernel-Module vmm und nmdm laden, tap (tunnel software network interface) für die VM und eine Brücke zwischen der echten PC-NIC (hier: Intel igb0) und dem tap einrichten.

Dazu eintragen:

/etc/sysctl.conf:

```
net.link.tap.up_on_open=1
```

/boot/loader.conf:

```
vmm_load="YES"
```

```
nmdm_load="YES"
```

```
if_bridge_load="YES"
```

```
if_tap_load="YES"
```

/etc/rc.conf:

```
cloned_interfaces="bridge0 tap0"
```

```
ifconfig_bridge0="addm igb0 addm tap0"
```

oder, bei mehreren taps für mehrere VMs:

```
cloned_interfaces="bridge0 tap0 tap1 tap2"
```

```
ifconfig_bridge0="addm re0 addm tap0 addm tap1 addm tap2"
```

Dann:

reboot

Detaillierte Anleitung:

<https://www.freebsd.org/doc/handbook/virtualization-host-bhyve.html>

Oracle Linux als Gast:

grub2-bhyve installieren:

pkg install sysutils/grub2-bhyve

oder:

```
cd /usr/local/sysutils/grub2-bhyve  
make install clean  
rehash
```

In /root/lug Verzeichnisse anlegen:

```
cd /zroot/lug  
mkdir dist  
mkdir oracle
```

Im neuen Verzeichnis "oracle" Image-Datei erstellen:

```
cd /zroot/lug/oracle  
truncate -s 32G oracle.img
```

Im gleichen Verzeichnis ein Geräteverzeichnis oracle.map anlegen. vi oracle.map, hier eingeben:

```
(hd0) /zroot/lug/oracle/oracle.img  
(cd0) /zroot/lug/dist/V100082-01.iso
```

In das neue Verzeichnis "dist" die Installation-DVD (iso, 4 GB) für Oracle Linux 7.2 hineinkopieren (Download: <https://edelivery.oracle.com>), Name:

V100082-01.iso

Bei den neuen Verzeichnissen und Dateien auf die Zugriffsrechte achten, für die Installation erhöhen (chmod 775), später ggf. wieder einschränken; auch die Besitzrechte genau einstellen (chown root:wheel).

Linux-Kernel von der iso mit grub-bhyve starten, hier mit 2048 RAM, Name der neuen VM ist oracle:

```
cd /zroot/lug/oracle  
grub-bhyve -m oracle.map -r cd0 -M 2048M oracle
```

Von der iso startet nun Grub. Für Ubuntu oder Debian sind (normalerweise) keine weiteren Einstellungen nötig, bei Oracle Linux muss man vmlinuz und initrd genau angeben:

```
grub> ls  
(hd0) (cd0) (cd0,msdos2) (host)
```

```
grub> ls (cd0)/  
Oracle_BuildTag EFI/ EULA GPL images/ isolinux/ LiveOS/ Packages/ repodata/ RPM  
-GPG-KEY-Oracle-Testing-7 RPM-GPG-KEY-Oracle-7 TRANS.TBL
```

```
grub> ls (cd0)/isolinux  
boot.cat boot.msg grub.conf initrd.img isolinux.bin isolinux.cfg memtest splash  
.png TRANS.TBL upgrade.img vesamenu.c32 vmlinuz
```

Kernel und Initrd laden:

```
grub> linux (cd0)/isolinux/vmlinuz  
grub> initrd (cd0)/isolinux/initrd.img
```

Installer nun starten:

```
grub> boot
```

Der Prompt erscheint wieder. Nun bhyve starten für die Installation von Oracle Linux, hier für eine Konfiguration mit zwei Kernen, dem tap0 und 2048 MB RAM. Die VM bekommt wieder den Namen "oracle":

```
bhyve -c 2 -m 2048M -H -P -A -l com1,stdio -s 0:0,hostbridge -s 1:0,lpc -s 2:0,virtio-  
net,tap0 -s 3,ahci-cd,/zroot/lug/dist/V100082-01.iso -s 4,virtio-  
blk,/zroot/lug/oracle/oracle.img oracle
```

Genaue Erklärung dieser und weiterer Parameter unter:

```
man bhyve
```

Der Anaconda-Installer startet nun im Textmodus. Oracle Linux einrichten, dabei nach Möglichkeit in /etc/ssh/sshd_config gleich einen passenden ssh-Zugang einrichten. Das neue Linux nach der Installation herunterfahren. Virtuelle Maschine wieder löschen:

```
bhyvectl --destroy --vm=oracle
```

Start-Skript:

Für die Oracle-Linux-VM nun Start-Skript erstellen (ausführbar machen: `chmod a+x`), dabei hier unbedingt für diese Linux-Variante das Verzeichnis von Grub auf dem Image eingeben (mit: `--directory=/grub2`).

Hier z. B. für 4 MB RAM, 4 Kernel, Netzwerkbrücke, `tap0`, Image `oracle.img`, zusätzlicher Zugang mit Null-Modem-Konsole (`nmdm`), Ampersand `&` am Ende für Unterdrückung der Bildschirmausgabe, damit die VM im headless-Modus betreiben werden kann:

```
#!/bin/sh
/usr/sbin/bhyvectl --destroy --vm=oracle
sleep 2
cd /zroot/lug/oracle
/usr/local/sbin/grub-bhyve -m oracle.map -r hd0,msdos1 --directory=/grub2 -M
4096M oracle
/usr/sbin/bhyve -AI -H -P -W \
  -s 0:0,hostbridge \
  -s 1:0,lpc -s 2:0,virtio-net,tap0 \
  -s 2:0,virtio-blk,/zroot/lug/oracle/oracle.img \
  -l com1,/dev/nmdm0A \
  -c 4 \
  -m 4096M oracle &
```

Die Einrückungen müssen stimmen und jede Zeile muss korrekt mit einem Zeilenumbruch enden; das z. B. bei `vi` mit `set:list` prüfen.

Einstellungen für eine real-live-Anwendung:

```
#!/bin/sh
/usr/sbin/bhyvectl --destroy --vm=oracle
sleep 2
cd /zroot/image/oracle/oracle
/usr/local/sbin/grub-bhyve -m oracle.map -r hd0,msdos1 --directory=/grub2 -M 4096M oracle
/usr/sbin/bhyve -AI -H -P -W \
  -s 0:0,hostbridge \
  -s 1:0,lpc \
  -s 2:0,virtio-net,tap2 \
  -s 3:0,virtio-blk,/zroot/image/oracle/oracle/oracle.img \
  -l com1,/dev/nmdm0A \
  -c 4 \
  -m 4096M oracle &
```

Genauere Erklärung der Parameter wie schon oben mit:

man bhyve

Hier finden sich auch weitere Konfigurationsbeispiele.

Weitere Optionen:

Verbindung zur VM mit Nullmodem-Konsole

(getestet, nmdm muss wie oben von bhyve gestartet werden)

Ersatz für Zugang über ssh bzw. mit einem Terminalprogramm wie tmux oder screen:

```
cu -l /dev/nmdm0B
```

<https://www.freebsd.org/doc/handbook/virtualization-host-bhyve.html>

Kapitel 21.7.5

ZFS-Block-Device ("volume") für bessere Performance statt einem Disk-Image wie oben

(nicht getestet)

(Beispiel: `zfs create -V32G -o volmode=dev zroot/linuxdisk0`)

<https://www.freebsd.org/doc/handbook/virtualization-host-bhyve.html>

Kapitel 21.7.4

bhyve für Windows Vista 7, 8, 10 und Windows Server 2008r2, 2012r2, 2016 Technical Preview

<https://wiki.freebsd.org/bhyve/Windows>

<https://jameslodge.com/freebsd-bhyve-hypervisor-running-windows-server-2012-r2-standard/>

<http://pr1ntf.xyz/windowsunderbhyve.html>

bhhyve mit PCI-Passthru für Netzwerkkarten

https://wiki.freebsd.org/bhyve/pci_passthru

Stand der Entwicklung von bhyve

<https://wiki.freebsd.org/bhyve>

Geplante Features

Live VM Migration

Support für VMDK, VHD und QCOW2

USB Pass-through

VirtFS Support

VM PXE Boot Support

VNC-Brücke

u. a.