Libvirt unter Gentoo

Grundsätzliches zu virsh

virsh {en} ist ein Werkzeug zum Verwalten virtueller Maschinen[6] unter Xen, QEMU, KVM, LXC, Test, OpenVZ, VirtualBox, OpenNebula oder VMware ESX. Man kann eine Maschine von der Kommandozeile aus erstellen, löschen, starten, stoppen und verwalten. Virsh ist für den fortgeschrittenen Linuxadministrator von Interesse, um einfache Skriptautomatismen für die Virtualisierung zu realisieren. Dieser Artikel ist besonders auf die Nutzung von virsh mit KVM ausgelegt. Wenn man sich bisher noch nicht mit Virtualisierung[6] beschäftigt hat oder bisher nur graphische Oberflächen wie die von Virtual-Box gewohnt ist und jetzt tiefer einsteigen möchte, sollte nach Möglichkeit erst mit der reinen KVM- oder QEMU-Technik auf der Konsole beginnen, um die Konfigurationsspezifika z.B. für eine Windowsmaschine kennen zu lernen.

Für einen stabilen Betrieb ist hier im Vorfeld einiges zu Themen, welche Virtuelle Netzwerkkarte oder welcher Festplattendateityp für den Fall das Beste sind[4][5], zu klären. Die Vorteile von virsh liegen dann im vereinfachten Verwalten der virtuellen Instanzen.

virsh ist ein Teil des Libvirt-Projekts {en}, das eine Library (~ Befehlssammlung) für diverse Programmiersprachen zur Verfügung stellt, mit denen die Hypervisoren gesteuert werden können. Es stellt gewissermaßen die Befehlssätze in der Shell zur Verfügung.

Konfiguration von Libvirt

```
* To allow normal users to connect to libvirtd you must change the
*  unix sock group and/or perms in /etc/libvirt/libvirtd.conf
*
* For the basic networking support (bridged and routed networks)
* you don't need any extra software. For more complex network modes
* including but not limited to NATed network, you'll need the
* following packages:
*
*      net-dns/dnsmasq
*      >=net-firewall/iptables-1.4.9
*      net-firewall/ebtables
```

KVM für System und User

If you don't have kvm compiled into the kernel, make sure you have the kernel module loaded before running kvm. The easiest way to ensure that the kernel module is loaded is to load it on boot. For AMD CPUs the module is called 'kvm-amd' For Intel CPUs the module is called 'kvm-intel' Please review /etc/conf.d/modules for how to load these.

```
*Make sure your user is in the 'kvm' group
```

```
* Just run 'gpasswd -a <USER> kvm', then have <USER> re-login.
*
* You will need the Universal TUN/TAP driver compiled into your
* kernel or loaded as a module to use the virtual network device
* if using -net tap.  You will also need support for 802.1d
* Ethernet Bridging and a configured bridge if using the provided
* kvm-ifup script from /etc/kvm.
*
* The gnutls use flag was renamed to ssl, so adjust your use flags.
```

Virtio Gastsupport

Im Kernel folgende zwei Module aktivieren:

```
Device Drivers --> Block devices --> Virtio block driver
Device Drivers --> Network device support --> Virtio network driver
```

From:
https://deepdoc.at/dokuwiki/ - **DEEPDOC.AT - enjoy your brain**

Permanent link:
https://deepdoc.at/dokuwiki/doku.php?id=gentoo:libvirt_unter_gentoo

Last update: **2025/11/29 22:06**