

Virtual Machine Live Migration

107021129 黃明灝

A. VM Setup Description

The experiments throughout this document were all done in a custom environment on my local machine. There are two “hosts”¹, analogous to host 1 and host 2 mentioned in the tutorial slides, which are themselves containers running under the LXC container runtime. The QEMU guest virtual machine to be live-migrated runs within these containers. To make things simple (and lightweight), both hosts and the guest are running the Alpine Linux distribution. The following steps were taken to setup the environment and the VM.

1. Launch containers.

```
1 lxc launch image:alpine:3.15 host1
2 lxc launch image:alpine:3.15 host2
```

2. Add a shared mount (where the VM disk image is to be put) and passthrough the KVM device to both hosts.

```
1 lxc storage volume create default nfs
2 lxc config device add host1 nfs disk \
3     pool=default source=nfs path=/mnt/nfs
4 lxc config device add host2 nfs disk \
5     pool=default source=nfs path=/mnt/nfs
6 lxc config device add host1 kvm unix-char path=/dev/kvm
7 lxc config device add host2 kvm unix-char path=/dev/kvm
```

3. Configure bridge and TAP interfaces in both hosts. To do so, edit `/etc/network/interfaces` like this.

```
1 auto tap0 inet manual
2     pre-up tuncctl -t tap0
3
4 auto br0
5 iface br0 inet dhcp
6     bridge-ports eth0 tap0
7     bridge-stp 0
8
9 hostname $(hostname)
```

Run `rc-service networking restart` to reload.

¹All uses of the word “host” in this document, unless otherwise specified, are referring to *host 1* and *host 2* running as containers on the actual host.

4. Launch QEMU to perform VM installation. Here we use `-curses` to show the TTY in a curses-based interface inside the terminal. This avoids the headaches of doing VNC and X11 forwarding².

```
1 qemu-system-x86_64 -cpu host -enable-kvm -m 1G -smp 1 \  
2   -drive if=virtio,format=raw,file=/mnt/nfs/alpine.img \  
3   -boot d -cdrom alpine-virt-3.15.4-x86_64.iso \  
4   -curses -nographic \  
5   -netdev tap,id=tap0,ifname=tap0,script=no,downscript=no \  
6   -device virtio-net-pci,netdev=tap0
```

Installation is straightforward:

- a. Login on TTY as `root`.
- b. Run `setup-alpine`.
- c. Accept the default settings all the way down. Set root password.
- d. Select `vda` as the target with the `sys` disk mode.
- e. Reboot.

Additionally, change `/etc/ssh/sshd_config` to allow root login.

5. The VM is now in good shape and ready to perform live migration experiments.

B. CPU Performance With and Without KVM Enabled

As background information, the data is obtained on a laptop running NixOS with a R7-4750U processor. Sysbench are ran for 10 seconds on all configurations.

- On guest **with** KVM enabled: 18678 events
- On guest **without** KVM enabled: 3621 events
- On host: 12378 events

The guest VM runs over 5 times slower with KVM disabled. This is caused by the fact that without KVM, the QEMU process has no access to the virtualization features provided by the kernel module, and thus having to rely on emulation by software. In contrast, the guest with KVM enabled performs almost identical to the host machine.

²My machine is on Wayland, which complicates X11 forwarding even more.