QEMU Networking Tips

There are a number of different mechanisms for tackling networking in *qemu* (I use *kvm* and *qemu* interchangeably).

Direct Bridge

A direct bridge gives *qemu* complete access to the network. This example is based on *Debian*.

Need 'tun' support

```
sudo modprobe tun
echo tun >> /etc/modules
```

Install tunctl and bring up the tunnel

```
sudo apt-get install uml-utilities
```

Setup /etc/network/interfaces

```
# The primary network interface
#iface eth0 inet dhcp

# qemu bridge
iface br0 inet dhcp
bridge_ports eth0
bridge_fd 1
bridge_hello 1
bridge_stp off
```

Setup /etc/qemu-ifup

```
#!/bin/sh

# bridge
echo "Executing /etc/qemu-ifup"
echo "Bringing up $1 for bridged mode..."
sudo /sbin/ifconfig $1 0.0.0.0 promisc up
echo "Adding $1 to br0..."
sudo /usr/sbin/brctl addif br0 $1
sleep 2
sudo chmod 755 /etc/qemu-ifup
```

Start the instance

```
sudo ifup br0
sudo tunctl -u john -t tun0

kvm -m 512 -hda hda.qcow2 -boot c -no-acpi -net nic,vlan=0 \
    -net tap,vlan=0,ifname=tun0,script=/etc/qemu-ifup
```

Standard qemu MAC prefix is 52:54:00.

Notes

Each instance started will need it's own *tun* and MAC address.

Using *Proxy Arp*

By using *proxy arp* you can bridge to a fixed IP address. One nice thing about using *proxy arp* is that you can bridge to a wireless NIC, which you cannot do with direct bridging as described above.

This method requires that the VM have static IPs.

Script

Instead of steps, here is a script, *kvm_networking*, to do the bridging:

```
#!/bin/bash

kvm_if_up(){
    tunctl -u john -t tap0

    sysctl net.ipv4.ip_forward=1
    sysctl net.ipv4.conf.wlan0.proxy_arp=1

    sysctl net.ipv4.conf.tap0.proxy_arp=1

    ip link set dev tap0 up

    route add -host 192.168.22.125 dev tap0
}

kvm_if_down(){

    sysctl net.ipv4.ip_forward=0
    sysctl net.ipv4.conf.wlan0.proxy_arp=0

    sysctl net.ipv4.conf.tap0.proxy_arp=0

    ip link set dev tap0 down

    tunctl -d tap0
}

if [[ $EUID -ne 0 ]]; then
  echo "This script must be run as root" 1>&2
  exit 1
else

    case "$1" in

        start)
```

```
            kvm_if_up
            ;;
        stop)
            kvm_if_down
            ;;
        *)
            echo "Usage: $0 {start|stop}"
            ;;
    esac

fi

exit 0
```

Start the instance

```
sudo kvm_networking start

kvm -m 512 -boot c -hda hda.qcow2 -no-acpi \
    -net nic,vlan=0,macaddr=52:54:00:00:00:01 \
    -net tap,ifname=tap0,script=no,vlan=0
```

Notes

- Replace *wlan0* with whatever interface you want to use.
- Replace 'john' with the user running the VM