SE 4455 – Cloud Computing

Virtualization: A Hobbyist 's Tale

Dr. Jagath Samarabandu

jagath@uwo.ca

TEB 351

519-661-2111 x80058

DIY Virtualization – Small/Medium

- Small scale (desktop, 2-3 VMs)
  - Parallels, VirtualBox, VMware player/workstation
  - Easy to get started
  - Typical scenario: multiple OSs on a single workstation, personal use
- Medium scale (server, 10-20 VMs)
  - QEMU, VMware workstation
  - Beginning to deal with scaling issues
  - Write own scripts to automate a few things

DIY Virtualization – Large Scale

- Large scale (server racks, 100's of VMs)
  - QEMU, VMware workstation etc. can still be used
  - OpenStack is beginning to look attractive
  - Proper management layer is essential
    - E.g. Libvirt
  - Use or adapt existing automation solutions

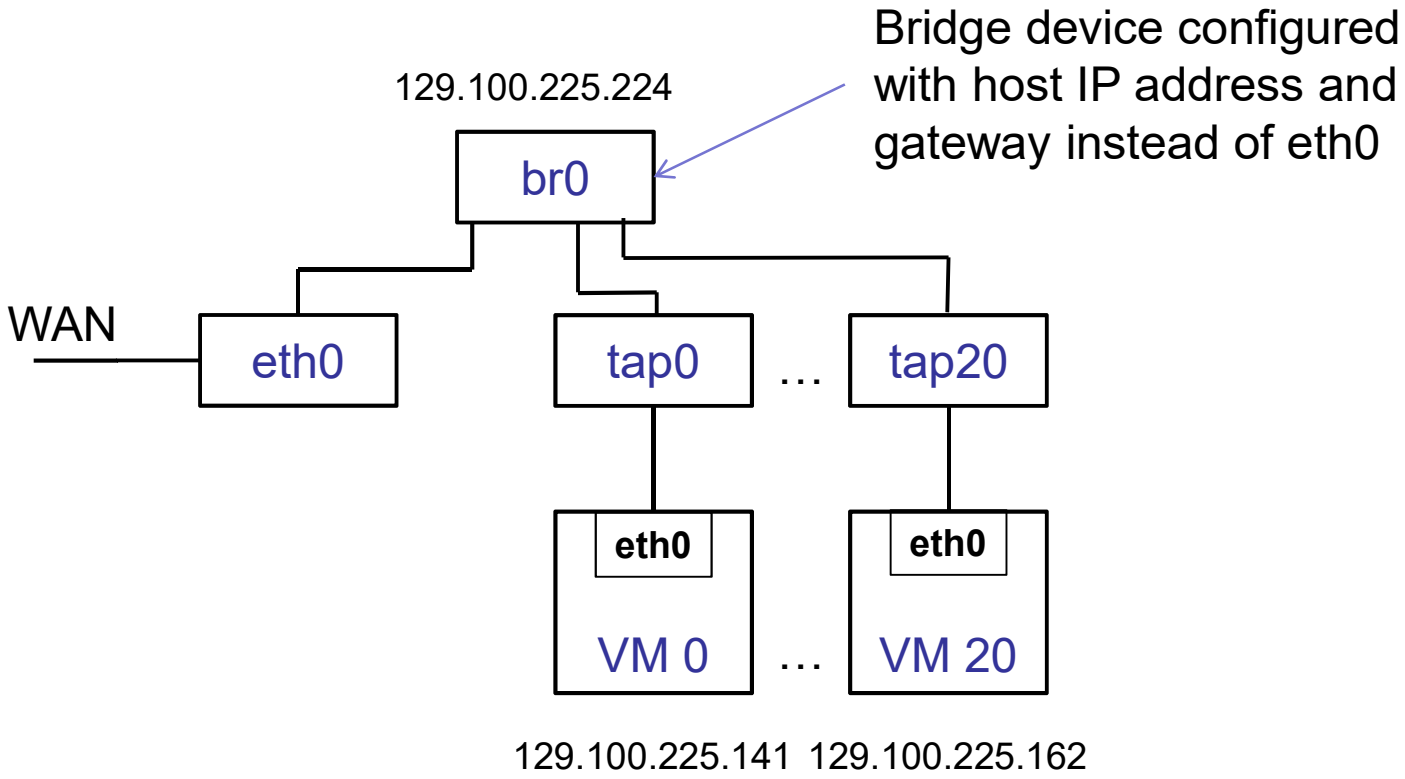
Case Study: Medium Scale

- Server
  - Two Intel Xeon E5520 CPUs (quad core, 2.27GHz), 48GB memory, 1TB disk
- Host OS
  - Gentoo Linux (Kernel version 4.4.39)
- Virtualization
  - QEMU with KVM
- Networking
  - One bridge (br0) with tap devices (tap0-tap20)

Linux Networking Terminology

- NIC: Network Interface Card – physical network card/port
- Network device: Each NIC is represented by Linux device
  - Listed in `/proc/net/dev` file (not in `/dev/`)
  - Use "`ifconfig -a`" to see all network devices
- Network bridge: Special network device created in software that acts as a switch
  - Use "`brctl addbr foo`" to create a bridge named "foo" (documentation: "`man brctl`")

Network Configuration



VM Preparation

- Disk image file
  - Format: QCOW2 – QEMU copy-on-write
  - Linux allows creating a file system within a file
- Base image created with Debian 8.7 net installer
  - Initial set of software installed on base image
  - Initial configuration done manually
- Each VM image only stores differences from this base image
- Second disk image used as a swap drive

Automation Scripts (Bash)

- All scripts written from scratch in 3-4 weeks (mostly figuring out how to use QEMU)
  - Hair tearing moment: figuring out how to boot install media in a text console
- Install OS: "qemu-install-os.sh"
- Start base image: "base-start.sh"
- Clone base image: "make_images.sh"
- Check if hosts are live: "check_hosts.sh"
- Start a given VM (or all): "vps_start.sh"

Networking

- Lot of trial and error and Google
- Two scripts to start and stop network interface for each VM
  - Invoked by QEMU

Customization Pain Points

- Setting hostname for each VM
  - Host name set manually for each VM image
  - Host DNS entry set manually
- Setting up DHCP
  - Forgot to set in base image. Had to do it by hand
- Issue with network names due to different MAC address in base vs clone

Automation Challenges

- Customizing each image with machine specific parameters
  - Host name/IP address
  - Account settings and keys
- Possible solution with qemu-nbd server
  - Serves blocks from a disk image as a network block device