Introduction to bhyve

John Baldwin

May 7, 2014

Overview

- What is bhyve?
- Requirements and Supported Guests
- Running a Guest
- Networking
- <Demo>

A Different Kind of Hypervisor

- Depends on Hardware Acceleration
- Varied Reasons
  - Expediency (x2APIC and MSI)
  - Sanity (EPT)
- No Firmware
  - BIOS (*sigh*)
  - UEFI / CSM
- Few Virtual Devices (VirtIO / AHCI)

Requirements

- Host CPU Support
  - Intel VT-x with EPT
    - “Unrestricted Guest” for SMP and i386
  - AMD SVM (AMD-V) with NPT in a Project Branch
    - http://mirrors.nycbug.org/pub/bhyve/
- Guest Support
  - VirtIO Drivers (or AHCI for Disk) (MSI)
  - Serial Console
  - Userspace Loader

Known Working Guests

- FreeBSD/amd64 8.x+
- FreeBSD/i386 8.x+
- OpenBSD/amd64 5.5 (not bsd.rd)
- NetBSD/amd64 (with some bhyve patches)
- Linux/x86-64 (various flavors)
- Illumos (with some hackish BIOS patches)

Components

- In-kernel Driver (vmm.ko)
- Userland Binary (bhyve(8))
- OS Loader
  - bhyveload(8) (FreeBSD)
  - sysutils/grub2-bhyve (everything else)
- bhyvectl(8)

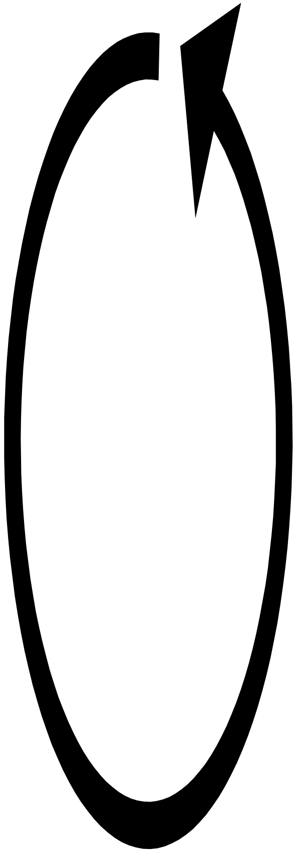
Guest Lifecycle

Three Steps:

1. bhyvectl –destroy

2. loader

3. bhyve <many args>



vmrun.sh

- /usr/share/example/bhyve/vmrun.sh
- Boots a FreeBSD guest using bhyveload(8) with a single VirtIO NIC and VirtIO disk
  - Second optional disk for installing
  - Various options
- ACPI soft-off breaks out of loop (SIGTERM)
- Decent template for your own scripts

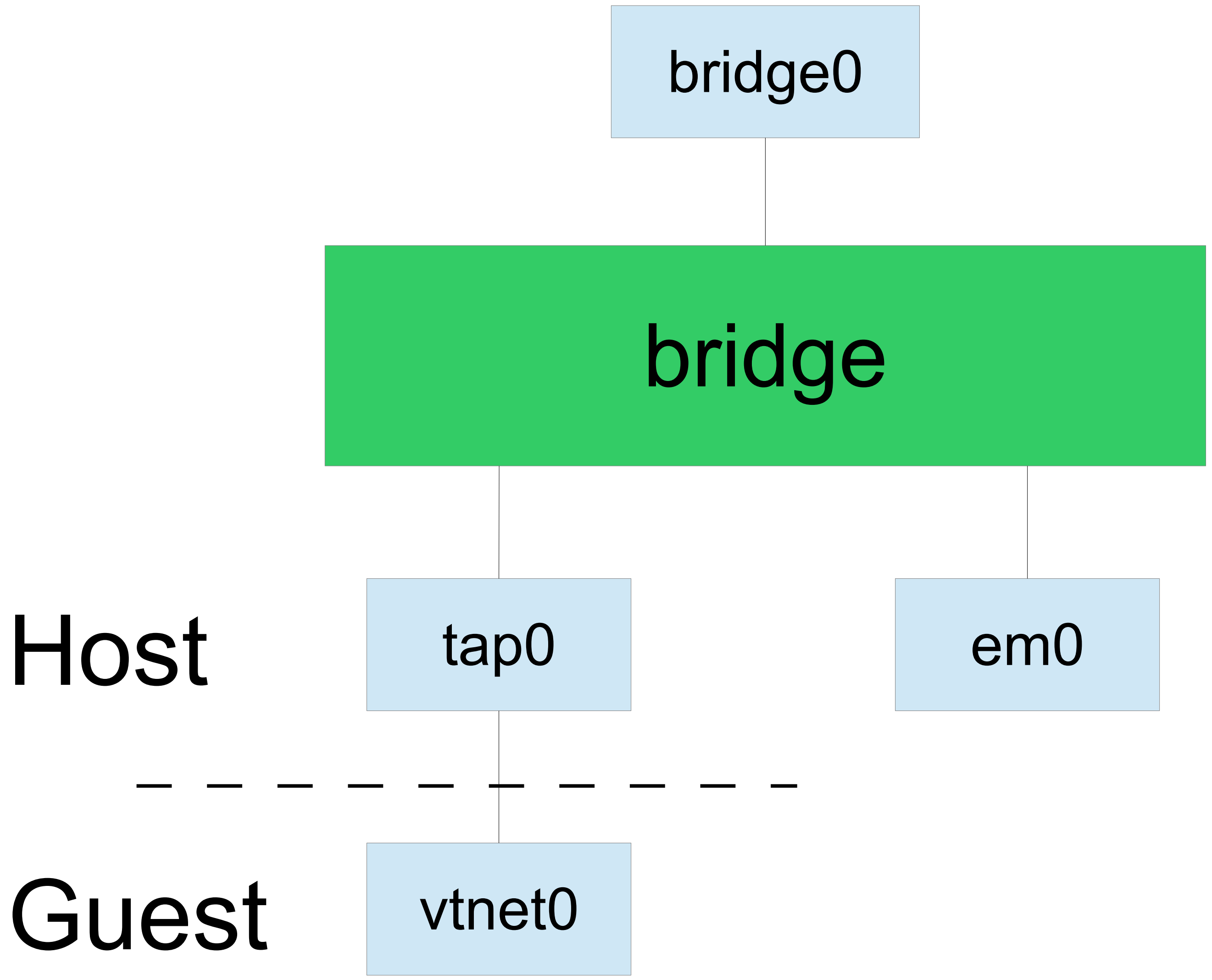
Detached Operation

- Serial console on COM1 uses stdio by default
- Option 1: tmux or screen
- Option 2: Attach console to nmdm(4) device
- SIGTERM for soft-off

Network Setup

- Allowing guests to get … somewhere
- Two sample setups
  - Bridged to a host-attached LAN
  - Use an internal LAN on the host
- Both setups use a bridge (if_bridge(4))
- Guest interfaces appear as tap(4) on the host

Bridged to Host-Attached LAN

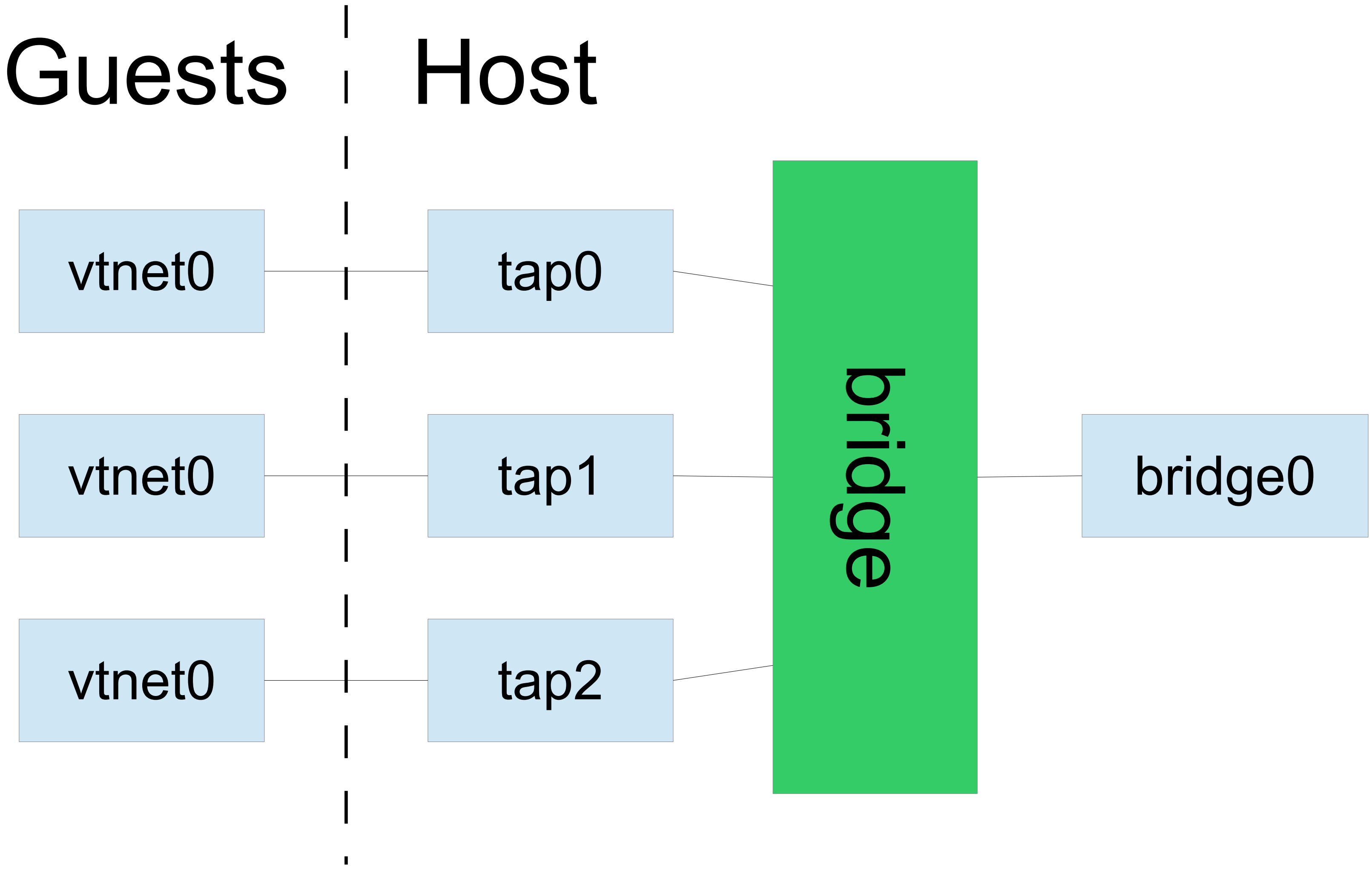


Bridged to Host-Attached LAN

```
# ifconfig bridge0 create
# ifconfig bridge0 addm em0
# ifconfig bridge0 addm tap0
# ifconfig bridge0 up
```

```
cloned_interfaces=”bridge0 tap0”
ifconfig_bridge0=”up”
autobridge_interfaces=”bridge0”
autobridge_bridge0=”em0 tap0”
```

Internal LAN Bridged to Host



Internal LAN Bridged to Host

```
cloned_interfaces=”bridge0 tap0 tap1 tap2”
ifconfig_bridge0=”inet 192.168.1.1/24”
autobridge_interfaces=”bridge0”
autobridge_bridge0=”tap*”
gateway_enable=YES
```

Internal LAN Bridged to Host

- NAT for outside access
  - Configure as if bridge0 was an interface to an internal LAN
- sysutils/dnsmasq makes life simpler
  - DHCP server for guests
  - DNS aliases for guests and host

Conclusion

- Demo
- Questions?

http://people.freebsd.org/~jhb/papers/bhyve/