Geelong

linux.conf.au 2016

Steven Ellis <sellis@redhat.com>
Senior Solution Architect
Red Hat NZ
IS THAT A DATACENTER IN YOUR POCKET?
IS THAT A CLOUD IN YOUR POCKET?
WHY?
I HAVE AN ITCH
I HAVE AN SCRATCH
I HAVE AN SCRATCH
I NEED TO ITCH
AGENDA

• Nested Virtualisation
• Thin LVM
• Snapshots
• RHEL-OSP vs RDO vs DevStack
• Atomic + cloud-init
AGENDA

• Along the way
  ▪ Choosing the right Base OS image(s)
  ▪ Gotchas
  ▪ Power Tips
AGENDA

- What Next?
  - UEFI
  - ARM64?
HARDWARE
fedora* 23

T440s 128 GB SSD

CentOS

Geelong

redhat.
DRAGONS
NESTING
NESTED VIRTUALISATION

Huge performance boost over qemu software emulation
NESTED VIRTUALISATION

Huge performance boost over qemu software emulation

/etc/modprobe.d/kvm-intel.conf

    options kvm-intel nested=1
    options kvm-intel enable_shadow_vmcs=1
    options kvm-intel enable_apicv=1
    options kvm-intel ept=1

OR

/etc/modprobe.d/kvm-amd.conf

    options kvm-amd nested=1

and reboot

redhat.
POWER TIP

Always use SSD Storage

Confirm your SSD Storage performance

- hdparm -f /dev/sdX
NESTED VIRTUALISATION

Create a dummy VM Centos7_BASE

- eg via virtual-manager

Edit Virtual Machine CPU type

virtsh edit Centos7_BASE

<cpu mode='custom' match='exact'>
  <model fallback='allow'>Nehalem</model>
  <vendor>Intel</vendor>
  <feature policy='require' name='vmx'/>
</cpu>
NESTING GOTCHA

Clashing default networks on 192.168.122.x

/etc/libvirt/qemu/networks/default.xml

<network>
  <name>default</name>
  <uuid>cf7022ad-7c12-4349-a15d-2bbcd3eba6ea</uuid>
  <bridge name="virbr0"/>
  <mac address='52:54:00:95:F9:9F'/>
  <forward/>
  <ip address="192.168.122.1" netmask="255.255.255.0">
    <dhcp>
      <range start='192.168.122.128' end='192.168.122.254' />
    </dhcp>
  </ip>
</network>
THIN LVM
Creating a thin pool on VG fedora_t440s

[root@t440s ~]# lvcreate -T -L 80G fedora_t440s/lv_thin
   Logical volume "lv_thin" created.

[root@t440s ~]# lvcreate -V40G -T fedora_t440s/thinpool -n Centos7_BASE
   Logical volume "Centos7_BASE" created.
LVM GOTCHAS

Thin on LVM
/etc/lvm/lvm.conf
issue_discards = 1

Thin on LVM on luks

- https://blog.christophersmart.com/2013/06/05/trim-on-lvm-on-luks-on-ssd
ALL ABOUT THAT BASE
BASE OS IMAGE

Doesn't have to be a minimal install

Have one for RHEL /Centos 6.x and 7.x (and Fedora / Ubuntu?)

Update your base regularly

How will you build your base image?
POWER TIP

ALWAYS USE A KICKSTART
BASE OS IMAGE

Doesn't have to be a minimal install

- What are your main use cases?
  - mine was OpenStack

- Don't skimp on disk
  - 40GB base HD

- Have some re-usable kickstart functions/snippets
# Rename VG2 as cinder-volumes in our post
# Kickstart doesn't like cinder-volumes as a VG name
#
# Only two partitions on disk as we're BIOS booting

part /boot --fstype=ext4 --size=300
part pv.253002 --size=30760
part pv.253003 --grow --size=1
volgroup vg_vml --pesize=4096 pv.253002
volgroup VG2 --pesize=4096 pv.253003
logvol / --fstype=ext4 --name=lv_root --vgname=vg_vml --grow --size=1024 --maxsize=51200
logvol swap --name=lv_swap --vgname=vg_vml --grow --size=2016 --maxsize=4032

redhat.
MINIMAL(ISH) PACKAGES

%packages
@core
@server-policy
ntp
@Virtualization-Platform
--firstboot
--NetworkManager
--NetworkManager-glib
mysql
# Install SSH keys.
install_ssh_keys()
{
    mkdir -p /root/.ssh
    chmod 700 /root/.ssh
    cat >> /root/.ssh/authorized_keys << KEYS
    ssh-rsa owkwdwdwmyreallysimpelpublickey steve@myhost
    KEYS
    chmod go-r /root/.ssh/authorized_keys
}
keep_yumcache()
{
    sed -i 's/keepcache=0/keepcache=1/g' /etc/yum.conf
}

fix_vg_name()
{
    vgrename VG2 cinder-volumes
}

install_ssh_keys
fix_vg_name
keep_yumcache

:redhat:
PATCH THAT BASE
POWER TIP

Keep a Content Cache

- rsync
  - nice and simple
POWER TIP

Keep a Content Cache

- rsync
  - nice and simple
- mrepo
  - More elegant
POWER TIP

Keep a Content Cache
  - rsync
    - nice and simple
  - mrepo
    - More elegant
  - RH Satellite 6.x / Katello + The Foreman
    - Double brownie points
    - Needs more storage
PATCH THE VM

# from my host environment
rsync -av 7/ centos7:/var/cache/yum/x86_64/7/

# From my VM
yum -y upgrade

# Pull changes back to my cache
rsync -av centos7:/var/cache/yum/x86_64/7/ 7/
ALL ABOUT THAT THIN BASE
MIGRATING TO THIN LVM

Power off your VM and get to know fstrim and kpartx

[root@t440s ~]# kpartx -a /dev/fedora_t440s/Centos7
[root@t440s ~]# cd /mnt; mkdir volume

[root@t440s mnt]# mount /dev/mapper/fedora_t440s-Centos7_BASE1 /mnt/volume/
[root@t440s mnt]# fstrim -v volume
boot: 356.5 MiB (373850112 bytes) trimmed
[root@t440s mnt]# umount volume/

[root@t440s mnt]# mount /dev/centos7/root /mnt/volume/
[root@t440s mnt]# fstrim -v ./volume
./boot: 9.8 GiB (10481520640 bytes) trimmed
[root@t440s mnt]# umount volume

[root@t440s mnt]# vgchange -a n centos7
  0 logical volume(s) in volume group "centos7" now active
[root@t440s mnt]# kpartx -d /dev/mapper/fedora_t440s-Centos7_BASE
LVTHIN GOTCHAS

virt-manager and thin

- Can't manage existing thin volumes
- You can manually add the path
LVTHIN GOTCHAS

virt-manager and thin

- Can't manage existing thin volumes
- **Don't** manually add the path
- **virsh edit <VM Image>**
  - manually enter the pathname
LVM THIN SNAPSHOTS
SNAPSHOTS OF SNAPSHOTS

- Great on Thin on SSD
  - minimal performance overhead

- I can easily create latest RDO or RHEL-OSP off my base

- Make snapshots as your layer things up

- Sensible names
  - Liberty_v1 vs Liberty_neutron_test

- Keep copious notes
THIN SNAPSHOT GOTCHA

# Thin no longer auto activates
# Thin no longer auto activates

# This is good

[root@t440s ~]# lvcreate -s --name Centos7_Liberty \ fedora_t440s/Centos7_BASE

Logical volume "Centos7_Liberty" created.

# Enable the thin volume
[root@t440s ~]# lvchange -ay -K fedora_t440s/Centos7_Liberty
LETS BUILD A CLOUD
# use virsh edit to re-point our VM at the new snapshot

# start VM

systemctl stop NetworkManager
yum remove NetworkManager\*
systemctl enable network
USE OUR CONTENT CACHE

# from my host environment

rsync -av 7/ \\  
centos:/var/cache/yum/x86_64/7/
RDO + PACKSTACK

Simple packstack - https://www.rdoproject.org/Quickstart

- yum -y upgrade
- yum install -y
- https://www.rdoproject.org/repos/rdo-release.rpm
- yum install -y openstack-packstack
- packstack --allinone --cinder-volumes-create=n

Uses our existing cinder-volumes as LVM backing for cinder

Rsync post packstack repo back to our host

- rsync -av centos:/var/cache/yum/x86_64/7/ /7/

Reboot and have a play
POWER TIP

Keep *digital* notes

- Pen + Paper = a start
- `vi` / `gedit` / `gnote`
- Google Docs / Google Keep
- Trac / Redmine / Wiki
- Document in Mojo
Ticket #193 (accepted task)

Create Centos 7 base image for demos

Reported by: steve
Priority: minor
Component: Centos
Keywords: 

Owned by: steve
Milestone: FY16Q4
Version: 

Needed for my LCA 2015 session to repeat some RDO and Atomic demos

Refs

- #167 Create local lv_thin pool on T440s for Demos
- #191 RHEL7 based OpenStack demo of RHELOSP8 Liberty
- #185 RHEL7 based OpenStack demo of RDO Kilo
- #159 RHEL7 based OpenStack demo of RHOS5
- RHEL7 based OpenStack demo of RHELOSP7 Kilo

Need to apply the following profile to the VM for nesting

```xml
<cpu mode='custom' match='exact'>
  <model fallback='allow'>Nehalem</model>
  <vendor>Intel</vendor>
  <feature policy='require' name='svm' />
</cpu>
```
FIREFALL GOTCHA

# Learn firewalld

firewall-cmd --permanent --add-service=http
firewall-cmd --permanent --add-service=https
firewall-cmd --permanent --add-port=5000/tcp
UBER POWER TIPS

Documentation and Repeatability

- I can blow away my snaps
Documentation and Repeatability

- I can blow away my snaps

Regularly patch base environment(s)
UBER POWER TIPS

Documentation and Repeatability

- I can blow away my snaps

Regularly patch base environment(s)

Re-image my base very quickly
EXPERIMENT

Lets do some neutron hacking.

[root@t440s ~]# lvcreate -s --name Centos7_Liberty_Net \ fedora_t440s/Centos7_Liberty
Logical volume "Centos7_Liberty_Net" created.

[root@t440s ~]# lvchange -a y -K fedora_t440s/Centos7_Liberty_Net

# Edit VM via virsh edit and restart
LVM VS QCOW
LEARN CLOUD-INIT

https://access.redhat.com/articles/rhel-atomic-cloud-init-faq

Create two data files we can inject via cloud-init

- meta-data
- user-data
EXAMPLE FILES

# cat >meta-data<<EOF

instance-id: Atomic0
local-hostname: atomic-00
EOF
EXAMPLE FILES

# cat >meta-data<<EOF
instance-id: Atomic0
local-hostname: atomic-00
EOF

# cat >user-data<<EOF
#cloud-config
password: atomic
chpasswd: {expire: False}
ssh_pwauth: True
sshAuthorizedKeys:
  - ssh-rsa <my rsa pubkey> steve@host
EOF
CREATE ISO IMAGE

genisoimage -output atomic0-cidata.iso \
-volid cidata -joliet -rock user-data meta-data

I: -input-charset not specified, using utf-8 (detected in locale settings)
Total translation table size: 0
Total rockridge attributes bytes: 331
Total directory bytes: 0
Path table size(bytes): 10
Max brk space used 0
183 extents written (0 MB)
QEMU + QCOW2

[root@t440s images]# qemu-img create -f qcow2 -o backing_file=../ISO/rhel-atomic-cloud-7.1-1.x86_64.qcow2 atomic-instance-0.qcow2

Formatting 'atomic-instance-0.qcow2', fmt=qcow2 size=10737418240 backing_file='../ISO/rhel-atomic-cloud-7.1-1.x86_64.qcow2' encryption=off cluster_size=65536 lazy_refcounts=off
# Use virt-install to create our first atomic host

# Note the cloud-init ISO file

virt-install --import --name Atomic0 --ram 2048 --vcpus 2 \--disk path=/opt/Virtual/images/atomic-instance-0.qcow2,format=qcow2,bus=virtio \--disk path=/opt/Virtual/images/atomic0-cidata.iso,device=cdrom \--network bridge=virbr0 --graphics vnc

# Lets start a console

virsh console Atomic0
POWER TIP

You can re-init the disk without re-running virt-install

- cloud-init cdrom will still be attached
- You can also snap your snap.
QCOW2 GOTCHAS

Don't snapshot a running instance
Playing with Docker and Kubernetes on Atomic

- mydbforweb
  - https://access.redhat.com/articles/1330533

- webwithdb
  - https://access.redhat.com/articles/1328953

- Getting started with Kubernetes
  - https://access.redhat.com/articles/1198103
SCRATCH I NEED TO ITCH
UEFI + KVM

Currently can boot a VM

Need to re-base

- VMs
- Kickstarts
- Then full Secure boot
Download RHEL 7.x 64bit ISO or Fedora

livecd-iso-to-disk \  
  --format --reset-mbr \  
  --efi rhel-server-7.1-x86_64-dvd.iso \  
  /dev/SDx

Boot USB in UEFI mode

  * force UEFI only in BIOS
STEVE'S WORLD OF CRAZY
UEFI + BIOS LIVE IMAGE

Live RHEL 7.x image

Can boot on BIOS and UEFI systems

Runs off my SD-CARD and USB3 card reader
RHEL 7.X INSTALL

In UEFI Mode

Leave a little space on the USB Key

- 8MB is sufficient

Boot / test / patch

Note your EFI layout

[root@rhel7sd 7Server]# parted /dev/sdd unit s print
Model: TS-RDF5 SD Transcend (scsi)
Disk /dev/sdb: 30539776s
Sector size (logical/physical): 512B/512B
Partition Table: gpt
Disk Flags:

<table>
<thead>
<tr>
<th>Number</th>
<th>Start</th>
<th>End</th>
<th>Size</th>
<th>File system</th>
<th>Name</th>
<th>Flags</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2048s</td>
<td>411647s</td>
<td>409600s</td>
<td>fat16</td>
<td>EFI System Partition</td>
<td>boot</td>
</tr>
<tr>
<td>2</td>
<td>411648s</td>
<td>1435647s</td>
<td>1024000s</td>
<td>xfs</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>1435648s</td>
<td>28706815s</td>
<td>27271168s</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
BIOS ENABLE

Legacy BIOS needs a couple of tweaks

- Additional bios_grub partition

```bash
# parted -a optimal /dev/sdd
(parted) unit s
(parted) mkpart non-fs 28706816s 28710911s

(parted) print
Model: TS-RDF5 SD Transcend (scsi)
Disk /dev/sdd: 30539776s
Sector size (logical/physical): 512B/512B
Partition Table: gpt
Disk Flags:

<table>
<thead>
<tr>
<th>Number</th>
<th>Start</th>
<th>End</th>
<th>Size</th>
<th>File system</th>
<th>Name</th>
<th>Flags</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2048s</td>
<td>411647s</td>
<td>409600s</td>
<td>fat16</td>
<td>EFI System Partition</td>
<td>boot, esp</td>
</tr>
<tr>
<td>2</td>
<td>411648s</td>
<td>1435647s</td>
<td>1024000s</td>
<td>xfs</td>
<td>msftdata</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>1435648s</td>
<td>28706815s</td>
<td>27271168s</td>
<td></td>
<td>lvm</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>28706816s</td>
<td>28710911s</td>
<td>4096s</td>
<td></td>
<td>non-fs</td>
<td></td>
</tr>
</tbody>
</table>

(parted) set 4 bios_grub on
```
# We'll only have grub2-efi installed

```
yum install grub2
```

Power off and force BIOS to boot **Legacy** only

Then boot from your Installer in rescue mode from USB Key or DVD-ROM
BIOS GRUB2

You **must** have booted in legacy mode to enable traditional grub2

chroot /mnt/sysimage
gerub-install --target=i386-pc /dev/sdx
gerub2-mkconfig -o /boot/grub/grub.conf

Reboot and test - should survive kernel upgrades
Retest in UEFI mode
NEXT?
TO DO

- Re-work all demos as UEFI ready kickstarts

- Kickstart the creation of my BIOS/UEFI live image
  - Add encryption support?

- Fully script Kubernetes on Atomic via cloud-init + Ansible
RED HAT® ENTERPRISE LINUX®
SERVER FOR ARM
Development Preview
THANK YOU

sellis@redhat.com
http://people.redhat.com/sellis

plus.google.com/+RedHat
linkedin.com/company/red-hat
youtube.com/user/RedHatVideos
facebook.com/redhatinc
twitter.com/RedHatNews

Geelong
linux.conf.au 2016