1 Gold Image Building

1.1 DiskImage Builder

- http://docs.openstack.org/developer/diskimage-builder/
- https://media.readthedocs.org/pdf/diskimage-builder/latest/diskimage-builder.pdf

---

Warning

- Distributions which are supported as a build host:
  - Centos 6, 7
  - Debian 8 (“jessie”)
  - Fedora 20, 21, 22
  - RHEL 6, 7
  - Ubuntu 14.04 (“trusty”)
  - Gentoo

- Distributions which are supported as a target for an image:
  - Centos 6, 7
  - Debian 8 (“jessie”)
  - Fedora 20, 21, 22
  - RHEL 6, 7
  - Ubuntu 12.04 (“precise”), 14.04 (“trusty”)
  - Gentoo

- For faster image building consider using a greater than 4GB of RAM instance flavor (m1.large).
  - You will get a error message like this when you don’t have enough memory for ‘fast-build’: *WARNING: Not enough RAM to use tmpfs for build. Using /tmp. (4046700 < 4G)*

1.1.1 Installation (Ubuntu)

1. Login on the new instance with your SSH key
    - sudo apt update
    - sudo apt install -y python-pip qemu-system kpartx python-glanceclient python3-glanceclient
    - sudo -H pip install diskimage-builder
    - sudo -H pip install --upgrade pip
    - sudo apt upgrade -y
    - Create an DIB folder in user home folder
        - cd
        - mkdir DIB
        - mkdir DIB/elements

Copy from controller1 the openrc file and put it in your user folder inside the Imagebuilder instance.

1.1.2 Define Elements

- cd DIB
- vi elements/02-citrix-centos
    - #!/bin/bash
      #
      echo "NOZEROCONF=yes" >> /etc/sysconfig/network
      cat > /etc/sysconfig/network-scripts/ifcfg-eth0 <<IFCFGETH0
      DEVICE="eth0"
      BOOTPROTO="dhcp"
      IPV6INIT="yes"
      MTU="1500"
      NM_CONTROLLED="no"
      ONBOOT="yes"
      TYPE="Ethernet"
      IFCFGETH0
      #
      sed -i -e 's/timeout=5/timeout=1\nserial --unit=0 --speed=115200\nterminal --timeout=10 console serial/g' /boot/grub/grub.conf
      sed -i -e 's#kernel /boot/vmlinuz-2\(.*\)#kernel /boot/vmlinuz-2\1 console=tty0 console=ttyS0,115200n8#g' /boot/grub/grub.conf
      dd if=/dev/urandom bs=1M count=1 2>/dev/null | sha512sum | awk '{print $1}' | passwd --stdin root
      #

- sudo mkdir /usr/local/share/diskimage-builder/elements/centos/post-install.d/

- sudo cp elements/02-citrix-centos /usr/local/share/diskimage-builder/elements/centos/post-install.d/
- sudo mkdir /usr/local/share/diskimage-builder/elements/centos7/post-install.d/
- sudo cp elements/02-citrix-centos /usr/local/share/diskimage-builder/elements/centos7/post-install.d/
- sudo mkdir /usr/local/share/diskimage-builder/elements/ubuntu/post-install.d/
- vi elements/02-citrix-ubuntu
    - #!/bin/bash
      #
      # ADAPT TO UBUNTU IF NEEDED
      #
      ##echo "NOZEROCONF=yes" >> /etc/sysconfig/network
      ##cat > /etc/sysconfig/network-scripts/ifcfg-eth0 <<IFCFGETH0
      ##DEVICE="eth0"
      ##BOOTPROTO="dhcp"
      ##IPV6INIT="yes"
      ##MTU="1500"
      ##NM_CONTROLLED="no"
      ##ONBOOT="yes"
      ##TYPE="Ethernet"
      ##IFCFGETH0
      #
      sed -i -e 's/timeout=5/timeout=1\nserial --unit=0 --speed=115200\nterminal --timeout=10 console serial/g' /boot/grub/grub.cfg
      sed -i -e 's#kernel /boot/vmlinuz-2\(.*\)#kernel /boot/vmlinuz-2\1 console=tty0 console=ttyS0,115200n8#g' /boot/grub/grub.cfg
      dd if=/dev/urandom bs=1M count=1 2>/dev/null | sha512sum | awk '{print $1}' | passwd root
      #
- sudo cp elements/02-citrix-ubuntu /usr/local/share/diskimage-builder/elements/ubuntu/post-install.d/

1.1.3 Build Image

- Citrix CentOS6 image
    - /usr/local/bin/disk-image-create --mkfs-options '-i 16384' -o dib-centos6 -p acpid,cloud-init,man,openssh-clients,rsync,augeas-libs,compat-db,compat-libstdc++-296,device-mapper-multipath,emacs-nox,ipa-client,iptraf,irqbalance,lftp,libpcap,libxml2-python,lynx,microcode_ctl,net-snmp,net-snmp-utils,python-devel,readahead,ruby,sendmail,sysfsutils,sysstat,vim-enhanced,yum-plugin-priorities,yum-utils centos vm

- Citrix CentOS7 image
    - /usr/local/bin/disk-image-create --mkfs-options '-i 16384' -o dib-centos7 -p acpid,cloud-init,man,openssh-clients,rsync,augeas-libs,compat-db,compat-libstdc++-296,device-mapper-multipath,emacs-nox,ipa-client,iptraf,irqbalance,lftp,libpcap,libxml2-python,lynx,microcode_ctl,net-snmp,net-snmp-utils,python-devel,readahead,ruby,sendmail,sysfsutils,sysstat,vim-enhanced,yum-plugin-priorities,yum-utils centos7 vm

- Ubuntu Xenial
    - Note on a bug:
        - If you have python not found errors at line 105
            - sudo vi /usr/bin/dib-run-part
                - Comment out lines 105 and 106
    - /usr/local/bin/disk-image-create --mkfs-options '-i 16384' -a amd64 -o dib-ubuntu-amd64 -p acpid,cloud-init,man,openssh-client,rsync,libdevmapper*,emacs24-nox,freeipa-client,iptraf,irqbalance,lftp,libpcap*,python-libxml2,lynx,microcode.ctl,net-snmp*,python-dev,ureadahead,ruby,sendmail,sysfsutils,sysstat,vim vm ubuntu

1.1.4 Upload image in Glance

- ssh to Disk Image Builder instance
- source openrc
- build_date=`date +%Y-%m-%d`
    - This could be part of a automated script
- glance image-create --name *image_name-dib-$build_date* --disk-format=qcow2 --container-format=bare --min-disk *size_in_gb* --min-*ram size_in_mb* --is-public *True* --progress < DIB/*image_file.qcow2*

    - CentOS 6 Image
        - glance image-create --name *centos-6-x86_64-cloud-dib-$build_date* --disk-format=qcow2 --container-format=bare --min-disk *4* --min-*ram 1024* --visibility *public* --architecture *x86_64* --progress < DIB/*dib-centos6.qcow2*

    - CentOS 7 Image
        - glance image-create --name *centos-7-x86_64-cloud-dib-$build_date* --disk-format=qcow2 --container-format=bare --min-disk *4* --min-*ram 1536* --visibility *public* --architecture *x86_64* --progress < DIB/*dib-centos7.qcow2*

    - Ubuntu Xenial Image
        - glance image-create --name *ubuntu_1604-x86_64-cloud-dib-$build_date* --disk-format=qcow2 --container-format=bare --min-disk *10* --min-*ram 2048* --visibility *private* --architecture *x86_64* --progress < DIB/*dib-ubuntu-amd64.qcow2*

1.2 Packer

- https://www.packer.io/docs/builders/openstack.html
- https://wiki.jenkins-ci.org/display/JENKINS/Packer+Plugin

1.2.1 Installation (Ubuntu)

2. Make sure you have the openrc file from a controller in your home folder
3. sudo apt update
4. sudo apt install -y unzip python-glanceclient python3-glanceclient qemu-utils
5. sudo apt upgrade -y
6. mkdir PACKER
7. mkdir PACKER/files
8. mkdir PACKER/glance
9. mkdir PACKER/json-files
10. mkdir PACKER/log
11. mkdir PACKER/shell-scripts
12. cd PACKER
13. https://www.packer.io/downloads.html
    - Get Linux 64 bit
    - Unzip packer zip file
14. vi ../.bashrc (add the following lines at the end)
    - #
    - export PACKER_LOG_PATH="/home/ubuntu/PACKER/log/packerlog.txt"
    - export PACKER_LOG=1
    - #
15. export PACKER_LOG_PATH="/home/ubuntu/PACKER/log/packerlog.txt"
16. export PACKER_LOG=1

1.2.2 Build Image

17. Ubuntu
    - From home folder
      - source openrc
      - cd PACKER
      - vi json-files/ubuntu-xenial.json
        - {

```
"builders": [{
  "type": "openstack",
  "source_image": "b6654be9-9602-445d-87cd-31c6229d5f50",
  "ssh_username": "ubuntu",
  "openstack_provider": "nova",
  "image_name": "ubuntu_1604-x86_64-cloud-packer",
  "networks": "7844b057-5f71-40f7-b1c2-01ec931ddcd3",
  "security_groups": "default",
```

```
      "flavor": "m1.medium"
    }],
    "provisioners": [
    {
    "type": "shell",
      "script": "/home/ubuntu/PACKER/shell-scripts/citrix_ubuntu.sh",
      "pause_before": "10s",
       "execute_command": "chmod +x {{ .Path }}; {{ .Vars }} sudo -E {{
  .Path }}"
    }
    ]
  }
```

- vi shell-scripts/citrix_ubuntu.sh
  - ```
    #!/bin/bash
    #
    # ADAPT TO UBUNTU IF NEEDED
    #
    ##echo "NOZEROCONF=yes" >> /etc/sysconfig/network
    ##cat > /etc/sysconfig/network-scripts/ifcfg-eth0 <<IFCFGETH0
    ##DEVICE="eth0"
    ##BOOTPROTO="dhcp"
    ##IPV6INIT="yes"
    ##MTU="1500"
    ##NM_CONTROLLED="no"
    ##ONBOOT="yes"
    ##TYPE="Ethernet"
    ##IFCFGETH0
    #
    sudo sed -i -e 's/timeout=5/timeout=1\nserial --unit=0
    --speed=115200\nterminal --timeout=10 console serial/g'
    /boot/grub/grub.cfg
    sudo sed -i -e 's#kernel /boot/vmlinuz-2\(.*\)#kernel /boot/vmlinuz-2\1
    console=tty0 console=ttyS0,115200n8#g' /boot/grub/grub.cfg
    sudo dd if=/dev/urandom bs=1M count=1 2>/dev/null | sha512sum | awk
    '{print $1}' | passwd root
    #
    sudo apt-get update -y
    sudo DEBIAN_FRONTEND=noninteractive apt-get install -yq acpid
    cloud-init man openssh-client rsync libdevmapper* emacs24-nox
    freeipa-client iptraf irqbalance lftp libpcap* python-libxml2 lynx
    microcode.ctl net-snmp* python-dev ureadahead ruby sendmail sysfsutils
    sysstat vim
    sudo apt-get upgrade -y
    ```
    ```

#

- chmod 755 shell-scripts/citrix_ubuntu.sh
- ./packer validate json-files/ubuntu-xenial.json
- ./packer build json-files/ubuntu-xenial.json

18. CentOS 6
    - From home folder
        - source openrc
        - cd PACKER
        - vi json-files/centos6.json
            - {
              "builders": [{
                "type": "openstack",
                "source_image": "09840f23-17fa-47fc-8731-d018cb210714",
                "ssh_username": "centos",
                "openstack_provider": "nova",
                "image_name": "centos-6-x86_64-cloud-packer",
                "networks": "7844b057-5f71-40f7-b1c2-01ec931ddcd3",
                "security_groups": "default",
                "flavor": "m1.medium"
              }],
              "provisioners": [
              {
              "type": "shell",
                "script": "/home/ubuntu/PACKER/shell-scripts/citrix_centos6.sh",
                "pause_before": "10s",
                 "execute_command": "chmod +x {{ .Path }}; {{ .Vars }} sudo -E {{ .Path }}"
              }
              ]
            }
    - vi shell-scripts/citrix_centos6.sh
        - #!/bin/bash
          #
          echo "NOZEROCONF=yes" >> /etc/sysconfig/network
          cat > /etc/sysconfig/network-scripts/ifcfg-eth0 <<IFCFGETH0
          DEVICE="eth0"
          BOOTPROTO="dhcp"
          IPV6INIT="yes"
          MTU="1500"
          NM_CONTROLLED="no"
          ONBOOT="yes"
          TYPE="Ethernet"

IFCFGETH0
#
sed -i -e 's/timeout=5/timeout=1\nserial --unit=0 --speed=115200\nterminal --timeout=10 console serial/g' /boot/grub/grub.conf
sed -i -e 's#kernel /boot/vmlinuz-2\(.*\)#kernel /boot/vmlinuz-2\1 console=tty0 console=ttyS0,115200n8#g' /boot/grub/grub.conf
dd if=/dev/urandom bs=1M count=1 2>/dev/null | sha512sum | awk '{print $1}' | passwd --stdin root
#
sudo yum install -y acpid cloud-init man openssh-clients rsync augeas-libs compat-db compat-libstdc++-296 device-mapper-multipath emacs-nox ipa-client iptraf irqbalance lftp libpcap libxml2-python lynx microcode_ctl net-snmp net-snmp-utils python-devel readahead ruby sendmail sysfsutils sysstat vim-enhanced yum-plugin-priorities yum-utils
sudo yum upgrade -y
#

- chmod 755 shell-scripts/citrix_centos.sh
- ./packer validate json-files/centos6.json
- ./packer build json-files/centos6.json

19. CentOS 7
    - From home folder
        - source openrc
        - cd PACKER
        - vi json-files/centos7.json
            - {
              "builders": [{
                "type": "openstack",
                "source_image": "02cfcc34-5f40-47ea-9588-d1473d69f326",
                "ssh_username": "centos",
                "openstack_provider": "nova",
                "image_name": "centos-7-x86_64-cloud-packer",
                "networks": "7844b057-5f71-40f7-b1c2-01ec931ddcd3",
                "security_groups": "default",
                "flavor": "m1.medium",
                "user_data_file" : "/home/ubuntu/PACKER/files/user-data.txt"
              }],
              "provisioners": [
              {
              "type": "shell",
                "script": "/home/ubuntu/PACKER/shell-scripts/citrix_centos7.sh",
                "pause_before": "10s",

```
        "execute_command": "chmod +x {{ .Path }}; {{ .Vars }} sudo -E {{
    .Path }}"
      }
      ]
    }
```

- vi files/user-data.txt
  - #!/bin/bash
    #
    sed -i -e '/Defaults requiretty/{ s/.*/# Defaults requiretty/ }' /etc/sudoers
    #
- vi shell-scripts/citrix_centos7.sh
  - #!/bin/bash
    #
    echo "NOZEROCONF=yes" >> /etc/sysconfig/network
    cat > /etc/sysconfig/network-scripts/ifcfg-eth0 <<IFCFGETH0
    DEVICE="eth0"
    BOOTPROTO="dhcp"
    IPV6INIT="yes"
    MTU="1500"
    NM_CONTROLLED="no"
    ONBOOT="yes"
    TYPE="Ethernet"
    IFCFGETH0
    #
    sed -i -e 's/timeout=5/timeout=1\nserial --unit=0 --speed=115200\nterminal --timeout=10 console serial/g' /boot/grub/grub.conf
    sed -i -e 's#kernel /boot/vmlinuz-2\(.*\)#kernel /boot/vmlinuz-2\1 console=tty0 console=ttyS0,115200n8#g' /boot/grub/grub.conf
    dd if=/dev/urandom bs=1M count=1 2>/dev/null | sha512sum | awk '{print $1}' | passwd --stdin root
    #
    sudo yum install -y acpid cloud-init man openssh-clients rsync augeas-libs compat-db compat-libstdc++-296 device-mapper-multipath emacs-nox ipa-client iptraf irqbalance lftp libpcap libxml2-python lynx microcode_ctl net-snmp net-snmp-utils python-devel readahead ruby sendmail sysfsutils sysstat vim-enhanced yum-plugin-priorities yum-utils
    sudo yum upgrade -y
    #
- chmod 755 shell-scripts/citrix_centos7.sh
- ./packer validate json-files/centos7.json
- ./packer build json-files/centos7.json

1.2.3 Convert for RAW to QCOW2

The *packer build* process builds a RAW Glance image which is 40GB avg. and we want to compress it into a QCOW2.

Example with a Ubuntu Glance Image:

1. source openrc
2. cd PACKER/glance
3. glance image-list | grep packer
    a. Find image id (.i.e.: 048f9115-74f6-4e24-963e-a166e619ae38)
4. Download RAW image from Glance
    a. build_date=`date +%Y-%m-%d`
        i. This could be part of a automated script
    b. glance image-download --progress --file ubuntu_1604-x86_64-cloud-packer-*$build_date*.raw *048f9115-74f6-4e24-963e-a166e619ae38*
5. Delete RAW image from Glance
    a. glance image-delete 048f9115-74f6-4e24-963e-a166e619ae38
6. Convert from RAW to QCOW2
    a. qemu-img convert -p -O qcow2 ubuntu_1604-x86_64-cloud-packer-*$build_date*.raw ubuntu_1604-x86_64-cloud-packer-*$build_date*.qcow2
7. Upload back into Glance
    a. glance image-create --name *ubuntu_1604-x86_64-cloud-packer-$build_date* --disk-format=qcow2 --container-format=bare --min-disk *40* --min-*ram 2048* --visibility *private* --architecture *x86_64* --progress < *ubuntu_1604-x86_64-cloud-packer-$build_date.qcow2*
8. Cleanup
    a. rm -f *ubuntu_1604-x86_64-cloud-packer-**