Ceph Configuration Files

Contents

Ceph Configuration Files

This page describes the configuration files that are required to deploy Ceph. The user needs to **manually** copy the following files in to the cloud definition (`~/helion-input/my_cloud/definition`) directory before starting the ceph deployment.

List of Configuration Files

The configuration files are as follows:

- *ring.yml*
- *baremetalConfig.yml*
- *ccp.yml*
- *disks_compute.yml*
- *disks_controller.yml*
- *disks_osd.yml*
- *interfaces_set_1.yml*
- *network_groups.yml*
- *net_global.yml*
- *servers.yml*
- *server_roles.yml*

ring.yml

The example file is as follows:

```
---
product:
    version: 2

ring-specifications:

    - region-name: region1
      rings:
        - name: account
          display-name: Account Ring
          server-network-group: OBJECT
          min-part-time: 24
          partition-power: 17
          replication-policy:
            replica-count: 3

        - name: container
          display-name: Container Ring
          min-part-time: 24
          server-network-group: OBJECT
          partition-power: 17
          replication-policy:
            replica-count: 3

        - name: object-0
          display-name: General
          default: yes
          min-part-time: 24
          server-network-group: OBJECT
          partition-power: 17
          replication-policy:
```

```
            replica-count: 3
```

baremetalConfig.yml

The values you enter in the `baremetal.yml` file are the details about each of the servers in your environment. The details of the values are obtained from the iLO console or from your vlan setup information.

The example file is as follows:

```
---
  product:
    version: 2
  baremetal_network:
    subnet: 192.168.50.0
    netmask: 255.255.255.0
    gateway: 192.168.50.1
    name_server: 16.110.135.51
    server_interface: eth1
  baremetal_servers:
    -
      node_name: controller-0001
      node_type: CCN-001-001
      pxe_mac_addr: 5C:B9:01:78:AE:48
      pxe_interface: eth0
      pxe_ip_addr: 192.168.50.10
      ilo_ip: 10.1.195.64
      ilo_user: Administrator
      ilo_password: password
    -
      node_name: controller-0002
      node_type: CCN-001-001
      pxe_mac_addr: 5C:B9:01:78:9E:E8
      pxe_interface: eth0
      pxe_ip_addr: 192.168.50.11
      ilo_ip: 10.1.195.65
      ilo_user: Administrator
      ilo_password: password
    -
      node_name: controller-0003
      node_type: CCN-001-001
      pxe_mac_addr: 5C:B9:01:78:7D:88
      pxe_interface: eth0
      pxe_ip_addr: 192.168.50.12
      ilo_ip: 10.1.195.66
      ilo_user: Administrator
      ilo_password: password
    -
      node_name: compute-0001
      node_type: CPN-001-001
      pxe_mac_addr: 5C:B9:01:78:7D:08
      pxe_interface: eth0
      pxe_ip_addr: 192.168.50.13
      ilo_ip: 10.1.195.67
      ilo_user: Administrator
      ilo_password: password
    -
      node_name: osd-0001
      node_type: CON-001-001
      pxe_mac_addr: 5C:B9:01:78:AE:08
      pxe_interface: eth0
      pxe_ip_addr: 192.168.50.14
```

```
      ilo_ip: 10.1.195.68
      ilo_user: Administrator
      ilo_password: password
    -
      node_name: osd-0002
      node_type: CON-001-001
      pxe_mac_addr: 5C:B9:01:78:7D:78
      pxe_interface: eth0
      pxe_ip_addr: 192.168.50.15
      ilo_ip: 10.1.195.69
      ilo_user: Administrator
      ilo_password: password
    -
      node_name: osd-0003
      node_type: CON-001-001
      pxe_mac_addr: 5C:B9:01:78:3F:D8
      pxe_interface: eth0
      pxe_ip_addr: 192.168.50.16
      ilo_ip: 10.1.195.70
      ilo_user: Administrator
      ilo_password: password
```

ccp.yml

The example file is as follows:

```
---
  product:
    version: 2

  control-planes:
    - name: ccp
      region-name: region-ccp
      common-service-components:
        - logging-producer
        - monasca-agent
        - stunnel
      clusters:
        - id: 3
          name: c1
          server-role: ROLE-CCP
          member-count: 3
          service-components:
            - ntp-server
            - swift-ring-builder
            - mysql
            - ip-cluster
            - apache2
            - keystone-api
            - keystone-client
            - rabbitmq
            - glance-api
            - glance-registry
            - glance-client
            - cinder-api
            - cinder-scheduler
            - cinder-volume
            - cinder-backup
            - cinder-client
            - nova-api
            - nova-scheduler
            - nova-conductor
```

```
            - nova-console-auth
            - nova-novncproxy
            - nova-client
            - neutron-server
            - neutron-ml2-plugin
            - neutron-l3-agent
            - neutron-dhcp-agent
            - neutron-metadata-agent
            - neutron-openvswitch-agent
            - neutron-client
            - horizon
            - swift-proxy
            - memcached
            - swift-account
            - swift-container
            - swift-object
            - swift-client
            - heat-api
            - heat-api-cfn
            - heat-api-cloudwatch
            - heat-engine
            - heat-client
            - openstack-client
            - ceilometer-api
            - ceilometer-collector
            - ceilometer-agent-central
            - ceilometer-agent-notification
            - ceilometer-expirer
            - ceilometer-common
            - ceilometer-client
            - zookeeper
            - kafka
            - vertica
            - storm
            - monasca-api
            - monasca-persister
            - monasca-notifier
            - monasca-threshold
            - monasca-client
            - logging-server
            - ops-console-web
            - ops-console-monitor
            - cmc-service
            - freezer-api
            - freezer-agent
            - ceph-monitor

      resource-nodes:
        - name: compute
          resource-prefix: compute
          server-role: ROLE-CPN
          service-components:
            - ntp-client
            - nova-kvm
            - nova-compute
            - neutron-l3-agent
            - neutron-metadata-agent
            - neutron-openvswitch-agent
            - neutron-lbaasv2-agent
            - freezer-agent

        - name: osd
          resource-prefix: ceph
```

```
          server-role: ROLE-OSD
          service-components:
            - ntp-client
            - ceph-osd
```

disks_compute.yml

The example file is as follows:

```
---
  product:
    version: 2

  disk-models:
  - name: DISK_SET_COMPUTE
    # two disk node; remainder of disk 1 used for volume group with two
 logical volumes
    # for /var/log and /var /crash
    # disk 2 is used to create second VG, used for nova compute
    volume-groups:
    # The policy is not to consume 100% of the space of each volume group.
    # 5% should be left free for snapshots and to allow for some
 flexibility.
      - name: hlm-vg
        physical-volumes:
         - /dev/sda_root
        logical-volumes:
          - name: root
            size: 65%
            fstype: ext4
            mount: /
          - name: log
            size: 15%
            mount: /var/log
            fstype: ext4
            mkfs-opts: -O large_file
          - name: crash
            size: 15%
            mount: /var/crash
            fstype: ext4
            mkfs-opts: -O large_file
```

disks_controller.yml

The value you enter in the `disk_controller.yml` are the storage name and the controller node type allocates to that storage.

The example file is as follows:

```
---
  product:
    version: 2

  disk-models:
  - name: DISK_SET_CONTROLLER
    # two disk node; remainder of disk 1 and all of disk 2 combined in
 single VG
    # VG is used to create three logical vols for /var, /var/log, and /var/
crash
    device-groups:
      - name: swiftobj
        devices:
```

```
          - name: /dev/sdb
          - name: /dev/sdc
        consumer:
          name: swift
          attrs:
            rings:
              - account
              - container
              - object-0
      - name: cinder-volume
        devices:
          - name: /dev/sdd
        consumer:
           name: cinder
    volume-groups:
    # The policy is not to consume 100% of the space of each volume group.
    # 5% should be left free for snapshots and to allow for some
 flexibility.
      - name: hlm-vg
        physical-volumes:
          - /dev/sda_root
        logical-volumes:
          - name: root
            size: 30%
            fstype: ext4
            mount: /
          - name: log
            size: 40%
            mount: /var/log
            fstype: ext4
            mkfs-opts: -O large_file
          - name: crash
            size: 10%
            mount: /var/crash
            fstype: ext4
            mkfs-opts: -O large_file
          - name: elasticsearch
            size: 10%
            mount: /var/lib/elasticsearch
            fstype: ext4
            mkfs-opts: -O large_file
          - name: zookeeper
            size: 5%
            mount: /var/lib/zookeeper
            fstype: ext4
            mkfs-opts: -O large_file
        consumer:
           name: os
```

disks_osd.yml

The example file is as follows:

```
---
  product:
    version: 2

  disk-models:
  - name: DISK_SET_OSD
    # two disk node; remainder of disk 1 and all of disk 2 combined in
 single VG
    # VG is used to create three logical vols for /var, /var/log, and /var/
crash
```

```
    device-groups:
      - name: ceph-osd-data-and-journal
        devices:
          - name: /dev/sdb
        consumer:
           name: ceph
           attrs:
             usage: data
             journal_disk: /dev/sdc
      - name: ceph-osd-data-and-shared-journal-set-1
        devices:
          - name: /dev/sdd
        consumer:
           name: ceph
           attrs:
             usage: data
             journal_disk: /dev/sdf
      - name: ceph-osd-data-and-shared-journal-set-2
        devices:
          - name: /dev/sde
        consumer:
           name: ceph
           attrs:
             usage: data
             journal_disk: /dev/sdf
    volume-groups:
    # The policy is not to consume 100% of the space of each volume group.
    # 5% should be left free for snapshots and to allow for some
  flexibility.
      - name: hlm-vg
        physical-volumes:
          - /dev/sda_root
        logical-volumes:
          - name: root
            size: 30%
            fstype: ext4
            mount: /
          - name: log
            size: 40%
            mount: /var/log
            fstype: ext4
            mkfs-opts: -O large_file
          - name: crash
            size: 10%
            mount: /var/crash
            fstype: ext4
            mkfs-opts: -O large_file
        consumer:
           name: os
```

The above sample file contains three OSD disks and two journal disks. One journal disk is shared by two OSD disks.

The disk model has the following fields:

device-groups	There can be several device groups. This allows different sets of disks to be used for different purposes.
name	This is an arbitrary name for the device group. The name must be unique.
devices	This is a list of devices allocated to the device group. We need to specify the list of devices needed for Ceph.

consumer	This specifies the service that uses the device group. A `name` field containing **ceph** indicates that the device group is used by Ceph.
attrs	This is the list of attributes.
usage	There can be several use of devices for a particular service. In the above sample, `usage` field contains **data** which indicates that the device is used for data storage.
journal_disk	It is to used to capture journal data. You can share the journal disk between two OSD disks.

Important: Minimum 3 OSD disks are required to configure Ceph.

interfaces_set.yml

The example file is as follows:

```
---
  product:
    version: 2

  interface-models:
    - name: INTERFACE_SET_1
      network-interfaces:

        - name: eth1
          device:
            name: eth1
          network-groups:
            - MGMT

        - name: eth2
          device:
            name: eth2
          network-groups:
            - EXTERNAL_VM
```

network_groups.yml

The value you enter in the `network_groups.yml` are network information of your environment.

The example file is as follows:

```
---
  product:
    version: 2

  network-groups:
    - name: MGMT
      hostname-suffix: mgmt
      hostname: true

      tags:
        - neutron.networks.vxlan
        - neutron.networks.vlan:
            provider-physical-network: physnet1
        - nova.compute.iscsi

      component-endpoints:
        - default
```

```
      routes:
        - default

      load-balancers:
        - provider: ip-cluster
          name: lb
          components:
            - default
          roles:
            - internal
            - admin

        - provider: ip-cluster
          name: extlb
#          external-name: mycloud.org
          components:
            - default
          roles:
            - public
          cert-file: my-public-cert



    - name: EXTERNAL_VM
      tags:
        - neutron.l3_agent.external_network_bridge
```

net_global.yml

The example file is as follows:

```
---
  product:
    version: 2

  networks:
    - name: NET_MGMT
      vlanid: 51
      tagged-vlan: false
      cidr: 192.168.51.0/24
      gateway-ip: 192.168.51.1
      network-group: MGMT

    - name: NET_EXTERNAL_VM
      vlanid: 52
      network-group: EXTERNAL_VM
```

servers.yml

Three servers are dedicated to OSD nodes. The IP address is assigned to each OSD nodes. You must enter the IP address of your environment to assign OSD nodes. They should also be in the `baremetalConfig.yml` file as the `pxe_ip_addr` value so you can copy them from there.

If you are going to be using network interface (nic) mapping then the definitions will exist in the `nic_mappings.yml` file but you will specify which value to use for each of your servers in this file.

Note: The entries in this file are examples, if you have more or less nodes you can copy and paste to include your entire environment.

In the following example file three servers are assigned as OSD nodes, which is the minimum requirement for Ceph deployment.

```
---
  product:
    version: 2

  servers:

    - id: ccn-0001
      ip-addr: 192.168.50.10
      role: ROLE-CCP

    - id: ccn-0002
      ip-addr: 192.168.50.11
      role: ROLE-CCP

    - id: ccn-0003
      ip-addr: 192.168.50.12
      role: ROLE-CCP

    - id: cpn-0001
      ip-addr: 192.168.50.13
      role: ROLE-CPN

    - id: con-0001
      ip-addr: 192.168.50.14
      role: ROLE-OSD

    - id: con-0002
      ip-addr: 192.168.50.15
      role: ROLE-OSD

    - id: con-0003
      ip-addr: 192.168.50.16
      role: ROLE-OSD
```

server_roles.yml

The example file is as follows:

```
---
  product:
    version: 2

  server-roles:

    - name: ROLE-CCP
      interface-model: INTERFACE_SET_1
      disk-model: DISK_SET_CONTROLLER

    - name: ROLE-CPN
      interface-model: INTERFACE_SET_1
      disk-model: DISK_SET_COMPUTE

    - name: ROLE-OSD
      interface-model: INTERFACE_SET_1
      disk-model: DISK_SET_OSD
```