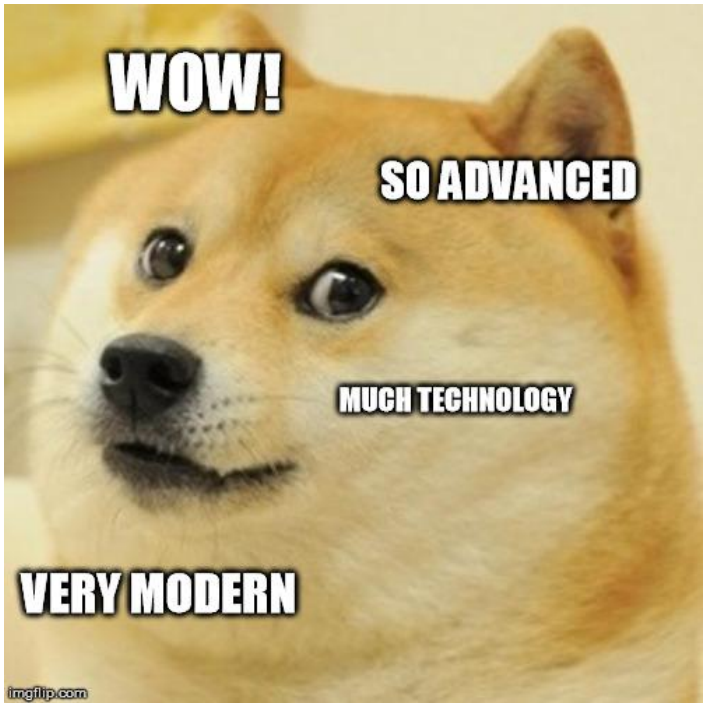
OpenStack deployment with Kolla Ansible and Ceph

+

+



Gigaset

Wherever you go.

OpenStack deployment with Kolla Ansible and Ceph

How to deploy and manage such big project as Openstack and Ceph?

- **Lots of options TripleO, ..., Devstack, using manual puppet/chef/ansible modules**

Gigaset

Wherever you go.

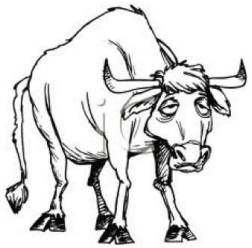
Openstack

Software platform for cloud computing, mostly deployed as an infrastructure-as-a-service (IaaS).
It started as joint project of Rackspace Hosting and NASA.
Latest stable release is Ocata

Use Cattle Not Pets

- Pets needs care and feeding
  - Name the VM
  - Tune and groom regularly
  - Feed pets with good food and supplements
  - Take pets to the vet when they are sick

- Cattle are disposable
  - Cloud instances are not unique
  - Tune and groom apps not the cattle
  - Replace when necessary
  - Shoot the cattle when it is sick



Gigaset

Wherever you go.

Ceph

Free-software storage platform, implements object storage on a distributed computer cluster. Made with enterprise features in mind to run on commodity hardware.

- **Scale out**
- **Data consistent (default replication 3)**
- **Most popular Cinder storage backend**
- **Space efficient thanks to CoW snapshots great. Instant deployment of new VMs if you use it for Glance image repository (enabled by default)**
- **Storage Tierieng with SSD/NVMe**

Ansible

Open-source automation engine that automates software provisioning, configuration management, and application deployment.[

- **agentless - no need to install additional daemons, firewall friendly,**
- **secure as your ssh keys**
- **Ansible uses Jinja2 templating to enable dynamic expressions and access to variables**
- **Low learning curve. Playbooks use an easy and descriptive language based on YAML**

Kolla

kollaó (gr κολλάω): to glue, unite

Kolla's mission is to provide production-ready containers (docker) and deployment tools (ansible) for operating OpenStack clouds. Kolla-Ansible is highly opinionated out of the box, but allows for complete customization. This permits operators with minimal experience to deploy OpenStack quickly and as experience grows modify the OpenStack configuration to suit the operator's exact requirements.

Kolla advantages/disadvantages

+ High availability OOB
+ docker containers
+ Ceph integration
+ full costumization of Openstack services
+ devops mindset
+ distribution of your choice CentOS, RHEL, Oracle Linux, Ubuntu

- docker containers
- slow deployment/reconfigure
- Openstack services defaults could be better

Links and screencasts

https://docs.openstack.org/developer/kolla-ansible/

Kolla Project - Deploying OpenStack in minutes
https://www.youtube.com/watch?v=9chL31_ViVI

Kolla Multi-node Deployment Tutorial - OpenStack Ocata
https://www.youtube.com/watch?v=8PTjeyOVScI

OpenStack Project Kolla Update
https://www.youtube.com/watch?v=ER5brUzM5R4
Tokyo Summit - Kolla: Ansible Deployment + OpenStack in Docker Containers = Operator Bliss
https://www.youtube.com/watch?v=BKYJuYsT4z4

IRC
#openstack-kolla (chat.freenode.net)
http://eavesdrop.openstack.org/irclogs/

Gigaset
Wherever you go.

Management VM preparation for kolla-ansible

Full procedure for CentOS/Ubuntu:
https://docs.openstack.org/developer/kolla-ansible/quickstart.html#install-dependencies

```
#CentOS
yum install epel-release
yum install python-pip
pip install -U pip

yum install ansible

yum install python-devel libffi-devel gcc openssl-devel

pip install kolla-ansible

cp -r /usr/share/kolla-ansible/etc_examples/kolla /etc/kolla/

cp /usr/share/kolla-ansible/ansible/inventory/multinode /etc/kolla/
```

Management VM preparation for kolla-ansible

Configure and install docker-engine >=1.10.0,!=1.13.0

```
#CentOS
curl -sSL https://get.docker.io | bash
yum install python-docker-py
```

Gigaset

Wherever you go.

Generate passwords

kolla-genpwd creates /etc/kolla/passwords.yml

```
cloudkitty_database_password: RtHSW1mWiafrfBousZ9AZl4tYuD1W1XW2q9zLBLC
cloudkitty_keystone_password: 874nSYzaSrqZjijn4Vc0tgT7OH9ummpOQaQi5MM7
congress_database_password: Isq4oOLITQwWBKzgz90c8TGzFqPXNTHRx30zGjI3
congress_keystone_password: m9yZnUCzNDVFFvQFoEAS1htU8Z8zUzFAQvQ16khW
database_password: zA5siPZ6LIy7BzhpuMIvzbGGDmpjdZxf6KAlhuMt
docker_registry_password: null
etcd_cluster_token: 1X9OyLuzwkTgTyQh6UAOTPbS2vltpqJShp1AMgHb
glance_database_password: bLHXdmX75g58xgwpAOkf643Nblu39UPuu1Pqt4lO
glance_keystone_password: V66k2SZd7GR8wGN9AIPwzpTCdQxlML56mbE9vfGm
gnocchi_database_password: OnmZ7UT6WmFuntHL6BkIsNQWTsrPzfSnyyub0061
gnocchi_keystone_password: ZqrZiBRJSIOrktGAQ5sGP5t4TApYTvkDOkcjUMpO
gnocchi_project_id: 82be5d1c-8407-47e8-bea7-9080b42e755e
gnocchi_resource_id: 2f68c613-408e-4c7d-afce-95fcce254b59
gnocchi_user_id: 3988bd9d-7d3c-433b-a391-db56d41e4534
grafana_admin_password: FeIvqVVdtIYNK5Py7JLr85vi2PKRYBXtMDO0WbFy
grafana_database_password: FtnAqBSJ6zrhLA9pYJt7NlhNd3SqXhpaBH9WxNnU
haproxy_password: NjDBGzzyb55fQuMdCise2h4mSgUUOZB5WTdBALbU
heat_database_password: o0kBvtDGuOp4QesD5u7TzQg8KSfcVNkgMHXgcdxR
heat_domain_admin_password: i1EOkTtR2dtVJRYWLLuKC2p63uIOZTEVbX3Dgvvf
heat_keystone_password: zf2krBGyHBn8Q4k9exXPvqwwgOk4d6mJNqxwNAVA
horizon_database_password: Lcz7R6Lk1PJR8UHiGWmvSId4PrzBOsyZra5Ow6o3
horizon_secret_key: H9tdnWxI8OjPeGRabsVY2lnLcdKjdcQHXv3ldpKb
ironic_database_password: B44z5jiCGdPcZmPKnOW378dpaBScUkiz36rD2Jdf
ironic_keystone_password: 7ZxuAUNMQvrQyqXBqdvbx8934o1Ez4zRWjpRjIyY
keepalived_password: bE7QhrAgf0eFpGItiJdkyMrw3N9Jv9mZ6BkLXZ28
keystone_admin_password: rPikRSIem7FwQCUzscw8BavPsx5wuaBHF7mV2Um3
keystone_database_password: N38h49tBDr6dhjbJrW8c2DQjittGAheFZU1pRgbZ
```

Do it only once!

Gigaset

Wherever you go.

Environment variables

/etc/kolla/globals.yml

```
####################
# Kolla options
####################
# Valid options are [ COPY_ONCE, COPY_ALWAYS ]
#config_strategy: "COPY_ALWAYS"

# Valid options are [ centos, fedora, oraclelinux, ubuntu ]
#kolla_base_distro: "centos"

# Valid options are [ binary, source ]
#kolla_install_type: "binary"

# Valid option is Docker repository tag
openstack_release: "3.0.0"
#openstack_release: "3.0.1"

# This should be a VIP, an unused IP on your network that will float between
# the hosts running keepalived for high-availability. When running an All-In-One
# without haproxy and keepalived, this should be the first IP on your
# 'network_interface' as set in the Networking section below.
kolla_internal_vip_address: "172.20.0.221"
#kolla_internal_vip_address: "172.29.1.12"

# This is the DNS name that maps to the kolla_internal_vip_address VIP. By
# default it is the same as kolla_internal_vip_address.
#kolla_internal_fqdn: "{{ kolla_internal_vip_address }}"

# This should be a VIP, an unused IP on your network that will float between
# the hosts running keepalived for high-availability. It defaults to the
# kolla_internal_vip_address, allowing internal and external communication to
# share the same address.  Specify a kolla_external_vip_address to separate
# internal and external requests between two VIPs.
#kolla_external_vip_address: "{{ kolla_internal_vip_address }}"

# The Public address used to communicate with OpenStack as set in the public_url
# for the endpoints that will be created. This DNS name should map to
# kolla_external_vip_address.
#kolla_external_fqdn: "{{ kolla_external_vip_address }}"

####################
# Docker options
####################
### Example: Private repository with authentication

docker_registry: "172.20.0.250:4000"
#docker_namespace: "companyname"
#docker_registry_username: "sam"
#docker_registry_password: "correcthorsebatterystaple"


:
```

Gigaset

Wherever you go.

Environment variables

/etc/kolla/multinode

```
# These initial groups are the only groups required to be modified. The
# additional groups are for more control of the environment.
[control]
# These hostname must be resolvable from your deployment host
wcm1.stack
wcm2.stack
wcm3.stack

# The above can also be specified as follows:
#control[01:03]     ansible_ssh_user=kolla

# The network nodes are where your l3-agent and loadbalancers will run
# This can be the same as a host in the control group
[network]
wcm1.stack

[compute]
won1.stack
won2.stack
won3.stack
won5.stack

[monitoring]
wcm2.stack

# When compute nodes and control nodes use different interfaces,
# you can specify "api_interface" and another interfaces like below:
#compute01 neutron_external_interface=eth0 api_interface=em1 storage_interface=em1 tunnel_interface=em1
won1.stack network_interface=bond1.200   cluster_interface=bond0.222 storage_interface=bond0.222  api_interface=bond1.200
won2.stack network_interface=bond1.200   cluster_interface=bond0.222 storage_interface=bond0.222  api_interface=bond1.200
won3.stack network_interface=bond1.200   cluster_interface=bond0.222 storage_interface=bond0.222  api_interface=bond1.200
won5.stack network_interface=bond1.200   cluster_interface=bond0.222 storage_interface=bond0.222  api_interface=bond1.200
wcm1.stack network_interface=enp5s0      storage_interface=enp9s0.222 api_interface=enp5s0
wcm2.stack network_interface=eno1        storage_interface=eno2.222   api_interface=eno1
wcm3.stack network_interface=enp5s0      storage_interface=enp9s0.222 api_interface=enp5s0
#mongodb1.stack network_interface=eth0
#mongodb2.stack network_interface=eth0
#mongodb3.stack network_interface=eth0
#ceilometer1.stack network_interface=eth0

[storage]
won1.stack
won2.stack
won3.stack
won5.stack

[gnocchivm]
wcm3.stack

[mgmtvm]
/etc/kolla/multinode
```

Gigaset

Wherever you go.

Building docker images with Openstack services

building your own repository using source/binary

kolla-build

kolla-build cinder

...few hours later

docker images

... or choose https://hub.docker.com/u/kolla/

kolla-ansible pull

Gigaset

Wherever you go.

Bootstraping servers

Baremetal preparation is done using:

```
kolla-ansible -i /etc/kolla/multinode bootstrap-servers
```

- Use btrfs for /var/lib/docker on destination hosts
- It's best to reboot servers after this operation completes
- You can easily add your own ansible code in kolla/ansible/roles/baremetal to deploy new baremetal faster

Deployment

kolla-ansible deploy -i /etc/kolla multinode

```
All Sensu checks green for this host.

[root@wcm2 STACK ~]# docker ps
CONTAINER ID        IMAGE                                                  COMMAND         CREATED             STATUS              PORTS               NAMES
f619bba76443        172.20.0.250:4000/kolla/centos-binary-ceph-rgw:3.0.0          "kolla_start"   12 days ago         Up 12 days                      ceph_rgw
e0723899f5b3        172.20.0.250:4000/kolla/centos-binary-ceilometer-api:3.0.0    "kolla_start"   9 weeks ago         Up 11 days                      ceilometer_api
1b1833a5d8c9        172.20.0.250:4000/kolla/centos-binary-mongodb:3.0.0           "kolla_start"   10 weeks ago        Up 9 weeks                      mongodb
a4c9a5ebdb3e        172.20.0.250:4000/kolla/centos-binary-horizon:3.0.0           "kolla_start"   3 months ago        Up 9 weeks                      horizon
bef412123dca        172.20.0.250:4000/kolla/centos-binary-cinder-scheduler:3.0.0  "kolla_start"   3 months ago        Up 9 weeks                      cinder_scheduler
d78543b12ebe        172.20.0.250:4000/kolla/centos-binary-cinder-api:3.0.0        "kolla_start"   3 months ago        Up 9 weeks                      cinder_api
7202707c1b66        172.20.0.250:4000/kolla/centos-binary-neutron-server:3.0.0    "kolla_start"   3 months ago        Up 2 weeks                      neutron_server
af53344c09b3        172.20.0.250:4000/kolla/centos-binary-nova-conductor:3.0.0    "kolla_start"   3 months ago        Up 9 weeks                      nova_conductor
07e580b7f5d4        172.20.0.250:4000/kolla/centos-binary-nova-scheduler:3.0.0    "kolla_start"   3 months ago        Up 9 weeks                      nova_scheduler
91551b6a7a5b        172.20.0.250:4000/kolla/centos-binary-nova-novncproxy:3.0.0   "kolla_start"   3 months ago        Up 9 weeks                      nova_novncproxy
c66172a47dbf        172.20.0.250:4000/kolla/centos-binary-nova-consoleauth:3.0.0  "kolla_start"   3 months ago        Up 9 weeks                      nova_consoleauth
6a1620c3135b        172.20.0.250:4000/kolla/centos-binary-nova-api:3.0.0          "kolla_start"   3 months ago        Up 9 weeks                      nova_api
70fd0c3da73b        172.20.0.250:4000/kolla/centos-binary-glance-api:3.0.0        "kolla_start"   3 months ago        Up 9 weeks                      glance_api
d91fb9b8d345        172.20.0.250:4000/kolla/centos-binary-glance-registry:3.0.0   "kolla_start"   3 months ago        Up 9 weeks                      glance_registry
0c640012ad04        172.20.0.250:4000/kolla/centos-binary-keystone:3.0.0          "kolla_start"   3 months ago        Up 9 weeks                      keystone
68c99d6f41da        172.20.0.250:4000/kolla/centos-binary-rabbitmq:3.0.0          "kolla_start"   3 months ago        Up 2 weeks                      rabbitmq
fa1d06a26f77        172.20.0.250:4000/kolla/centos-binary-memcached:3.0.0         "kolla_start"   3 months ago        Up 9 weeks                      memcached
1576037ef644        172.20.0.250:4000/kolla/centos-binary-elasticsearch:3.0.0     "kolla_start"   3 months ago        Up 9 weeks                      elasticsearch
97d8f628f204        172.20.0.250:4000/kolla/centos-binary-mariadb:3.0.0           "kolla_start"   3 months ago        Up 2 weeks                      mariadb
e1b3012ae38a        172.20.0.250:4000/kolla/centos-binary-ceph-mon:3.0.0          "kolla_start"   3 months ago        Up 9 weeks                      ceph_mon
eadb3a01aeba        172.20.0.250:4000/kolla/centos-binary-cron:3.0.0              "kolla_start"   3 months ago        Up 9 weeks                      cron
d6a6ddf1fb92        172.20.0.250:4000/kolla/centos-binary-kolla-toolbox:3.0.0     "kolla_start"   3 months ago        Up 9 weeks                      kolla_toolbox
dd84ca980be6        172.20.0.250:4000/kolla/centos-binary-heka:3.0.0              "kolla_start"   3 months ago        Up 2 weeks                      heka
[root@wcm2 STACK ~]#
```

Gigaset

Wherever you go.

Deployment

```
[root@won2 STACK ~]# docker ps
CONTAINER ID        IMAGE                                                                COMMAND         CREATED           STATUS          PORTS               NAMES
e4f5334e1ffc        172.20.0.250:4000/kolla/centos-binary-ceilometer-compute:3.0.0       "kolla_start"   10 weeks ago      Up 10 weeks                         ceilometer_compute
582d5438c484        172.20.0.250:4000/kolla/centos-binary-cinder-backup:3.0.0            "kolla_start"   11 weeks ago      Up 11 weeks                         cinder_backup
594566df2b4e        172.20.0.250:4000/kolla/centos-binary-cinder-volume:3.0.0            "kolla_start"   11 weeks ago      Up 11 weeks                         cinder_volume
22df2c32843f        172.20.0.250:4000/kolla/centos-binary-ceph-osd:3.0.0                 "kolla_start"   12 weeks ago      Up 8 weeks                          ceph_osd_17
6236f960aa7c        172.20.0.250:4000/kolla/centos-binary-ceph-osd:3.0.0                 "kolla_start"   12 weeks ago      Up 8 weeks                          ceph_osd_16
0d5984cfa36e        172.20.0.250:4000/kolla/centos-binary-ceph-osd:3.0.0                 "kolla_start"   12 weeks ago      Up 8 weeks                          ceph_osd_15
04883234e47e        172.20.0.250:4000/kolla/centos-binary-ceph-osd:3.0.0                 "kolla_start"   12 weeks ago      Up 8 weeks                          ceph_osd_14
97a5e7ed0978        172.20.0.250:4000/kolla/centos-binary-ceph-osd:3.0.0                 "kolla_start"   12 weeks ago      Up 8 weeks                          ceph_osd_13
a948d3195eed        172.20.0.250:4000/kolla/centos-binary-neutron-linuxbridge-agent:3.0.0 "kolla_start"   12 weeks ago      Up 2 weeks                          neutron_linuxbridge_
cefc7c44a0ee        172.20.0.250:4000/kolla/centos-binary-nova-ssh:3.0.0                 "kolla_start"   12 weeks ago      Up 12 weeks                         nova_ssh
4ea07f61e23b        172.20.0.250:4000/kolla/centos-binary-nova-compute:3.0.0             "kolla_start"   12 weeks ago      Up 12 weeks                         nova_compute
c3c34b361cda        172.20.0.250:4000/kolla/centos-binary-nova-libvirt:3.0.0             "kolla_start"   12 weeks ago      Up 12 weeks                         nova_libvirt
1badc251231e        172.20.0.250:4000/kolla/centos-binary-cron:3.0.0                     "kolla_start"   12 weeks ago      Up 12 weeks                         cron
759806ef4ea1        172.20.0.250:4000/kolla/centos-binary-kolla-toolbox:3.0.0            "kolla_start"   12 weeks ago      Up 12 weeks                         kolla_toolbox
b87d00ed0bc9        172.20.0.250:4000/kolla/centos-binary-heka:3.0.0                     "kolla_start"   12 weeks ago      Up 12 weeks                         heka
[root@won2 STACK ~]#
```

Gigaset

Wherever you go.

Reconfiguration

Kolla allows the operator to override configuration of services. Kolla will look for a file in /etc/kolla/config/<< service name >>/<< config file >>. This can be done per-project, per-service or per-service-on-specified-host. For example to globally change nova scheduler, the operator needs to create /etc/kolla/config/nova/nova-scheduler.conf
If the operator wants to reconfigure specific host - myhost, the operator needs to create file /etc/kolla/config/nova/myhost/nova.conf

```
[root@kolla-deploy STACK kolla]# cat /etc/kolla/config/nova/nova-compute.conf
[DEFAULT]
dhcp_domain = ""

[libvirt]
# Needed for ceph live migration http://docs.ceph.com/docs/master/rbd/rbd-openstack/#havana-and-icehouse
libvirt_live_migration_flag="VIR_MIGRATE_UNDEFINE_SOURCE,VIR_MIGRATE_PEER2PEER,VIR_MIGRATE_LIVE,VIR_MIGRATE_PERSIST_DEST,VIR_MIGRATE_TUNNELLED"
# Virtual CPU downgrade instead of real Intel/AMD models so live migration works
cpu_mode=custom
cpu_model=kvm64
volume_use_multipath = True
[root@kolla-deploy STACK kolla]#
```

kolla-ansible reconfigure -i /etc/kolla/multinode -t nova

Reconfiguration

To prepare an OSD as a storage drive, execute the following operations:

parted /dev/sdb -s -- mklabel gpt mkpart KOLLA_CEPH_OSD_BOOTSTRAP 1 -1

The following shows an example of using parted to configure /dev/sdb for usage with Kolla.

parted /dev/sdb -s -- mklabel gpt mkpart KOLLA_CEPH_OSD_BOOTSTRAP 1 -1

parted /dev/sdb print

Model: VMware, VMware Virtual S (scsi)

Disk /dev/sdb: 10.7GB

Sector size (logical/physical): 512B/512B

Partition Table: gpt

Number	Start	End	Size	File system	Name	Flags
1	1049kB	10.7GB	10.7GB		KOLLA_CEPH_OSD_BOOTSTRAP	

External Journal on SSD drive:

parted /dev/sdc -s -- mklabel gpt mkpart KOLLA_CEPH_OSD_BOOTSTRAP_FOO_J 1 -1

Deploy ceph

kolla-ansible deploy -i /etc/kolla/multinode -t ceph

Tips

- **bootstrap-server - don’t do it again on production system**
- **whenever you’d like to add new server ie compute node prepare new ansible inventory file for kolla. Using this file you can bootstrap-server only one server**
- **you can test your new configuration on openstack control/compute nodes by changing /etc/kolla/{openstack-service}/{configurationfile} ie /etc/kolla/cinder-api/cinder.conf and restartin proper docker container but every kolla-ansible reconfigure will reset it state**
- **all logs and data from containers is located on hosts in /var/lib/docker/volumes**
- **monitor ceph cluster from inside container $docker exec ceph_mon ceph -w**

Contact

alteriks@gmail.com

Presentation link:

Gigaset

Wherever you go.