Contrail as the CNI for Red Hat OpenShift

Last edited by **Simon Green** just now

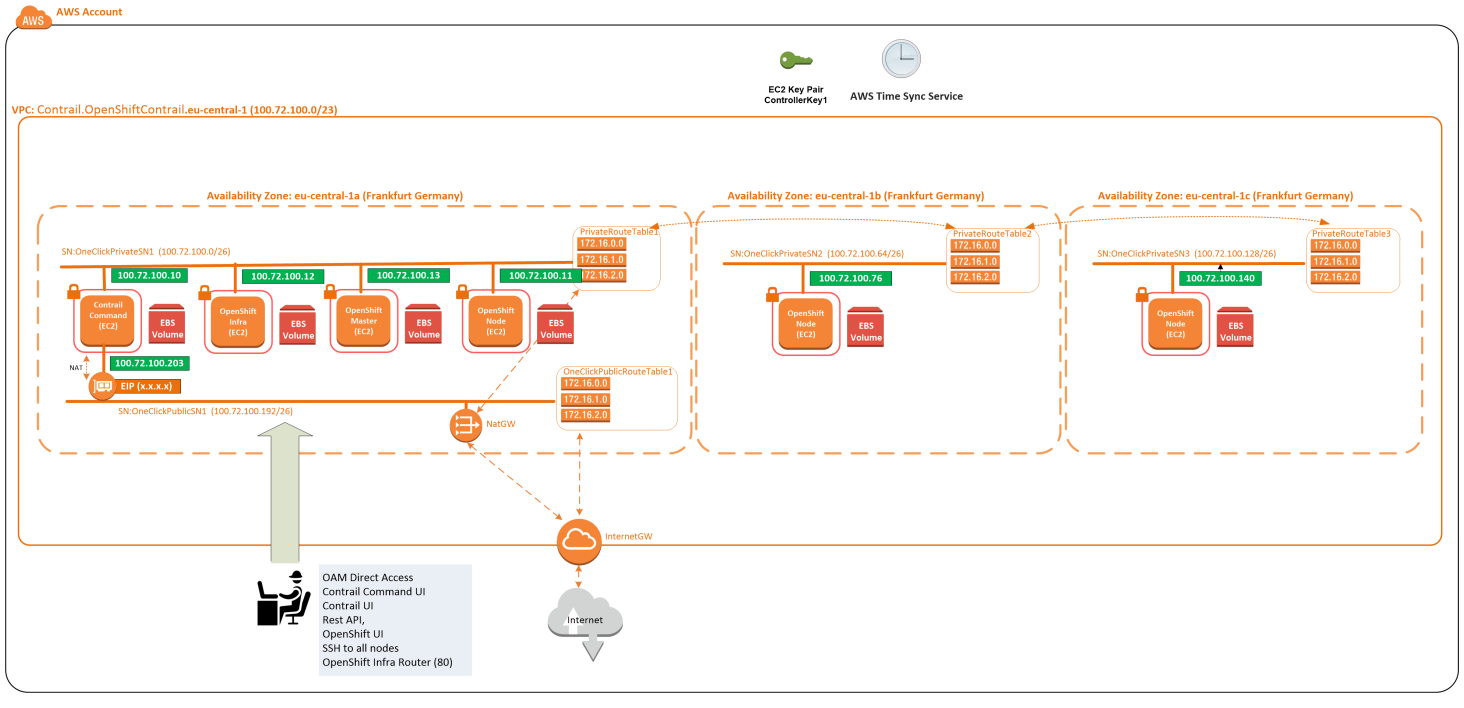
Deploy an HA Contrail SDN controller cluster across three AZs

---

There are four planned sections of this wiki.

1. Greenfield deployment, non HA, (small): deployer all infra plus OpenShift, Contrail and Contrail-Command (beta)
2. Brownfield deployment, non HA, (small): deploy OpenShift, Contrail and Contrail-Command into your existing VPC (beta)
3. Deploy in HA deployment using the same stacks, flag HA=True (pending, code in progress)
4. Importing the OpenShift cluster into Contrail Command

Greenfield OpenShift+Contrail deployment, non HA.



This stack and procedure deploys the VPCs, gateways, networks, security. Then within deploys and configures a non HA cluster of OpenShift with Contrail SDN as the CNI. Contrail Command for management and access.

One Time Setup:

1. You need an AWS account. Login to the AWS console with your browser.
2. In your AWS account console you need to an EC2 SSH key pair named "ContrailKey" within the region Frankfurt
3. You may need to accept the RHEL ami, the stack will fail and tell you that if its required.
4. You will need an account on the juniper contrail docker registry hub.juniper.net/contrail (user and password) in order to use our contrail SDN software.

How to deploy:

1. Click this link to launch the CloudFormation stack into your account (Frankfurt) Launch Stack

   Note: if you right click and copy this link, then share it with your customers via email. It will work for them as well
   ->NEXT
   complete the field ContainerRegistryUserName (this is your juniper docker registry user name)
   complete the field ContainerRegistryPassword (this is your juniper docker registry password)
   complete the field RedHatAccountUserName (This is the Red Hat account to register against for OpenShift)
   complete the field RedHatAccountPassword (This is the Red Hat account password to register against for OpenShift)
   complete the field RedHatSubscriptionPoolID (This is the pool id to use in that Red Hat account for OpenShift)
   strongly suggest changing the field UserLocation (by default 0.0.0/0 will make contrail command assessable to internet, set it to your laptop ip x.x.x.x/32 to lock it down)
   ->NEXT
   ->NEXT
   ->Deploy Stack

Note: All fields, can be changed, as can the region. However for the first run we suggest you go with the defaults.
Note: For production deployments you should change the flavour from m4.2xlarge to m4.4xlarge

1. Get a coffee, it typically takes 6 minutes for the CloudFormation stack to build, then another 45 minutes for the Openshift Ansible deployer to complete.
2. Once the stack completes the CloudFormation outputs tab will show you how to access the hosts and Contrail Command UI.
3. To watch the deployment, ssh to the Infra node, "tail -f /var/log/bootstrap.log"

Post Installation Steps:

Once the deployer has completed:

The admin and UI login and passwords are now handled in stack. OpenShift is configured with the Contrail Password set in the stack parameters.

Note: you might want to "oc adm [node name] --schedulable=false" to make the infra nodes non schedulable

To access the Contrail UI over internet, use the link in outputs. Default login is admin contrail123.

To access the OpenShift UI over internet, use the Link provided in the stack outputs. As we use private hostnames in the stack, unless you have the opportunity to add public DNS entries, on your Mac/pc you should add the following entry to /etc/hosts.

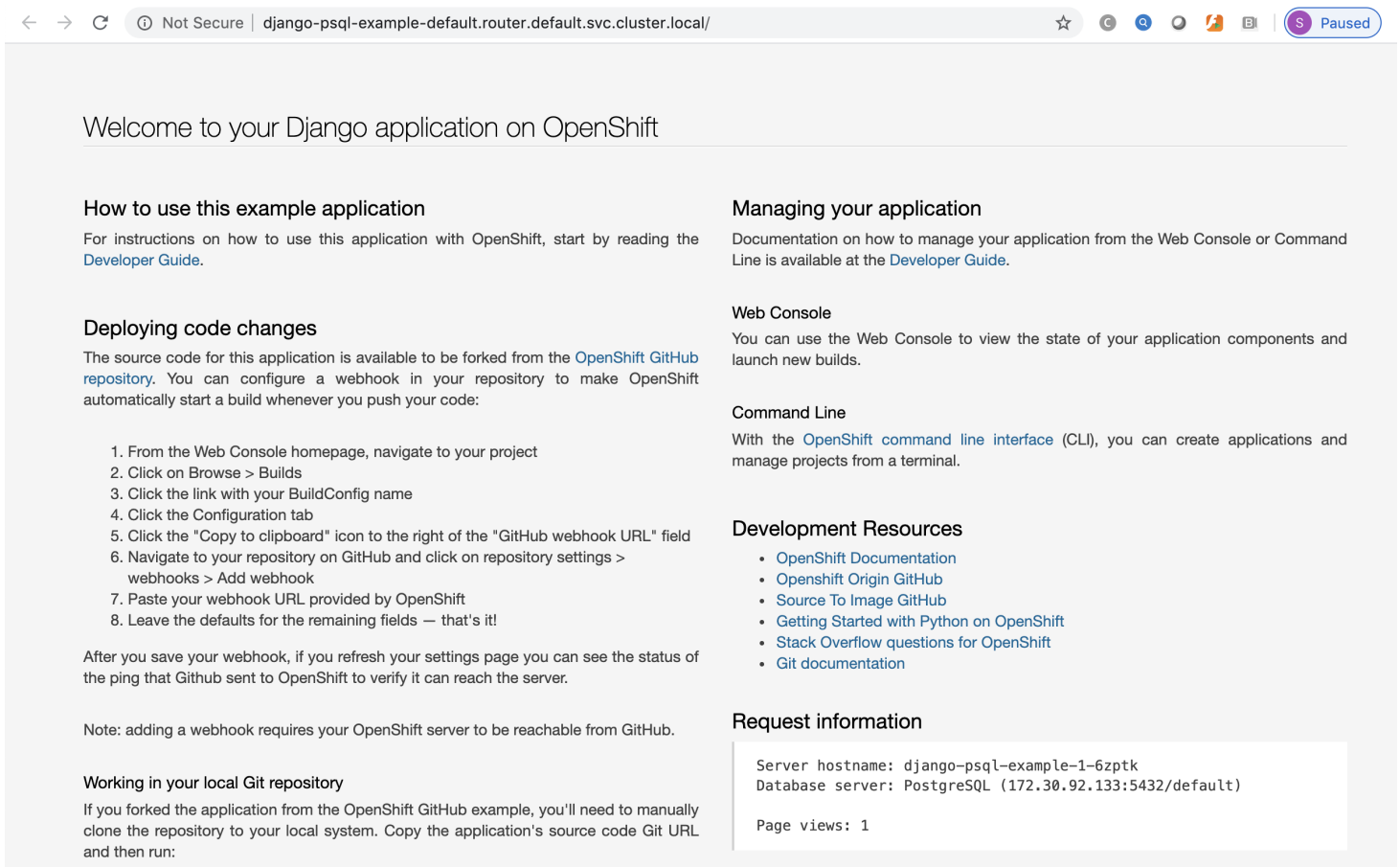
/etc/hosts
[public ip of contrail command] [fqdn of a master node]

for example, my /etc/hosts
3.126.125.30 ip-100-72-100-11.eu-central-1.compute.internal
3.126.125.30 ip-100-72-100-12.eu-central-1.compute.internal
3.126.125.30 ip-100-72-100-13.eu-central-1.compute.internal
3.126.125.30 django-psql-example-default.router.default.svc.cluster.local
3.126.125.30 alertmanager-main-openshift-monitoring.router.default.svc.cluster.local
3.126.125.30 grafana-openshift-monitoring.router.default.svc.cluster.local
3.126.125.30 prometheus-k8s-openshift-monitoring.router.default.svc.cluster.local

Note: The last three dns entries are for the monitoring platform endpoints, these come from "oc -n openshift-monitoring get routes" and with https:// can be used to reach the integrated prometheus/grafana monitoring platform.

Running a quick test OpenShift application:

1. Open the Openshift UI (see stack outputs for the link)
2. Select the Default project (top right)
3. Select the Catalog
4. Select the Django+PostgreSQL(Ephemeral) app. Add a password. Deploy it.
5. Click overview (top left) You will now see an http link for your new app at the top. For example. http://django-psql-example-default.router.default.svc.cluster.local
6. Unless you setup a real DNS domain and fqdn for your apps (see stack parameter OpenShiftAppDNSSubDomain). Add this FQDN to your Mac/pc /etc/hosts (see above example), again point to the public address of Contrail Command.
7. Wait 2 minutes for the app to come up.
8. Point your browser to the http shown in the UI and you will see your web app.
9. This also works for https based apps such as Jenkins.



What does this AWS stack build:

1. AWS infrastructure across three availability zones, including OAM access over internet
2. Contrail Command
3. A Contrail command server which is built and configured.
4. An OpenShift infra server, used to deploy the Ansible and to host Contrail and OpenShift infra services.
5. An OpenShift Master server which is built and configured.
6. Three OpenShift nodes (workers) in three AZs, build and configured.

How to secure OAM access:

1. The stack parameter UserLocation determines who can connect to it.
2. If you leave it blank we will allow access from 0.0.0.0/0, the whole of the internet.
3. If you set it to your laptop ip address x.x.x.x/32 only you can connect to it, its secured within the public SG.
4. You can also manually change the rules in the contrail command AWS security group.

How to enable access over a gateway to on premise:

1. The stack parameters SGSubnet1, SGSubnet2, SGSubnet3, SGSubnet4 are the subnets allowed to contact the instances from on premise (over your VPC gateway).
2. SGSubnet1 has a default, it can be changed, however has to be populated.
3. SGSubnet2,3,4 if left blank will not be populated in the SGs.

What Next:

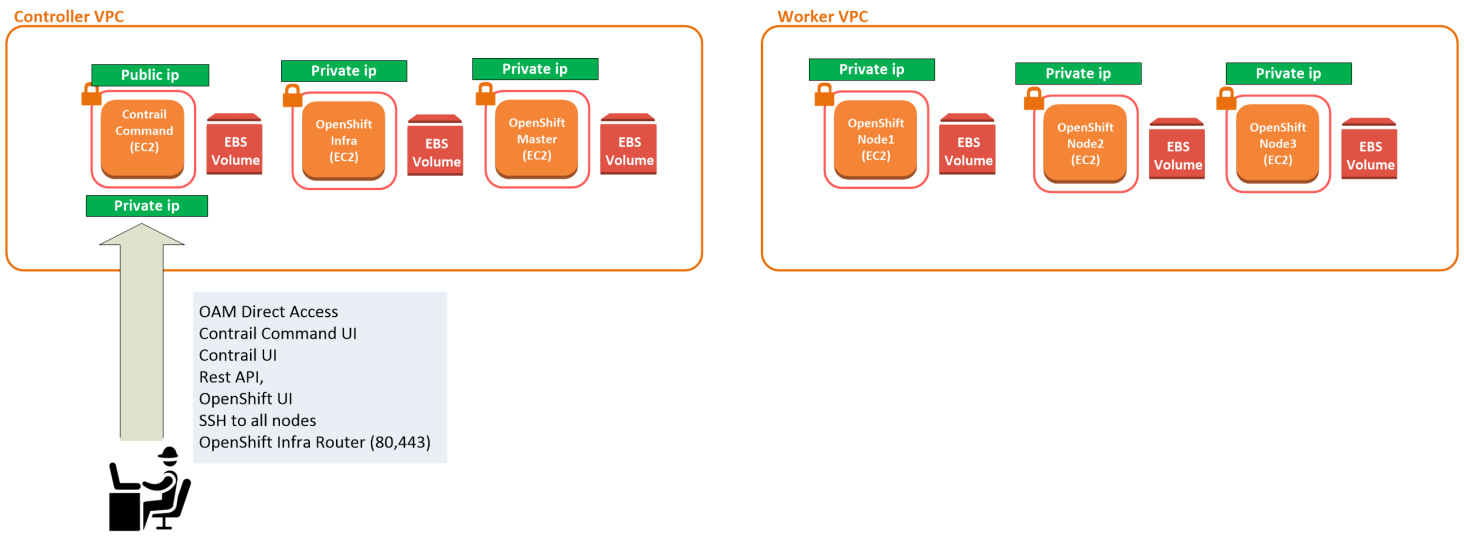
1. Brownfield stack
2. Finish off the HA.

Brownfield OpenShift+Contrail deployment, non HA.

note: HA is not ready yet so please leave the switch set to false

This is a fairly complex deployment, so we recumbent you start with the Greenfield stack above first.

This stack uses existing VPC's, Subnets and Gateways, and deploys the instances into them
This stack is designed to deploy the controllers (Master, Infra, Contrail-Command) into one VPC and subnets.
The nodes (workers) into another VPC and subnets.
If you specify the Same VPC's and Subnets then the instances will all end up deployed together, which also works.



There are some requirements on the infrastructure we deploy into.

1. The VPC needs the following options enabled: EnableDnsHostnames: True and EnableDnsSupport: True
2. The instances require accesss to the public internet in order to download software, so a NAT Gateway connected to the private subnets.
3. The public subnet used by the Contrail Command Instance requires an internet gateway, so that it can be accessed.
4. All instances require access to each other, wherever they may sit.
5. The Workers can be deployed into separate subnets or into the same subnet.
6. The controllers and the workers can be deployed into separate VPC's or the same VPC's.
7. use "oc adm manage-node --selector=region=infra --schedulable=false" to make infra nodes non schedulable

One Time Setup:

1. You need an AWS account. Login to the AWS console with your browser.
2. In your AWS account console you need to an EC2 SSH key pair named "ContrailKey" within the region Frankfurt
3. You may need to accept the RHEL ami, the stack will fail and tell you that if its required.
4. You will need an account on the juniper contrail docker registry hub.juniper.net/contrail (user and password) in order to use our contrail SDN software.

How to deploy:

1. Click this link to launch the CloudFormation stack into your account (Frankfurt) Launch Stack

   Note: if you right click and copy this link, then share it with your customers via email. It will work for them as well
   ->NEXT
   complete the field ContainerRegistryUserName (this is your juniper docker registry user name)
   complete the field ContainerRegistryPassword (this is your juniper docker registry password)
   complete the field RedHatAccountUserName (This is the Red Hat account to register against for OpenShift)
   complete the field RedHatAccountPassword (This is the Red Hat account password to register against for OpenShift)
   complete the field RedHatSubscriptionPoolID (This is the pool id to use in that Red Hat account for OpenShift)
   complete the field idVPCCONTROLLERS (VPCid you wish to place the Master, Infra and Contrail-Command Instances into)
   complete the field idContrailCommandPublicSubnet1 (Public Subnet used by the Contrail Command Instance for access)
   complete the field idControllersPrivateSubnet1 (private subnet you wish to place the Controller into)
   complete the field idControllersPrivateSubnet2 (any dummy value as we are not ha xxxxx)
   complete the field idControllersPrivateSubnet3 (any dummy value as we are not ha xxxxx)
   complete the field idVPCWORKERS (VPC you wish to place the worker nodes in to)
   complete the field idWorkersPrivateSubnet1 (Subnet to place the first worker instance into)
   complete the field idWorkersPrivateSubnet2 (Subnet to place the second worker instance into, can be the same)
   complete the field idWorkersPrivateSubnet3 (Subnet to place the third worker instance into, can be the same)
   complete the field ContrailCommand1PublicIP (Public Subnet IP Address for Contrail Command)
   complete the field ContrailCommand1PrivateIP (Private Subnet IP address for Contrail Command)
   complete the field OpenShiftInfra1PrivateIP (Private Subnet IP address for the Infra Node)
   complete the field OpenShiftMaster1PrivateIP (Private Subnet IP address for the Master Node)
   complete the field OpenShiftNode1PrivateIP (Private Subnet IP address for the first worker node)
   complete the field OpenShiftNode2PrivateIP (Private Subnet IP address for the second worker node)
   complete the field OpenShiftNode3PrivateIP (Private Subnet IP address for the third worker node)

strongly suggest changing the field UserLocation (by default 0.0.0/0 will make contrail command assessable to internet, set it to your laptop ip x.x.x.x/32 to lock it down)
->NEXT
->NEXT
->Deploy Stack

As this stack requires a lot of parameters, you may prefer to use the AWS cli. Here is an example.

```
----------------------------------------
aws cloudformation create-stack \
--stack-name OpenShift-Brownfield \
--template-url https://s3-eu-central-1.amazonaws.com/contrail-one-click-deployers/Contrail-OpenShift-Brownfield.json \
--parameters \
ParameterKey=ContainerRegistryUserName,ParameterValue="xxxxx" \
ParameterKey=ContainerRegistryPassword,ParameterValue="xxxxx" \
ParameterKey=RedHatAccountUserName,ParameterValue="xxxxx" \
ParameterKey=RedHatAccountPassword,ParameterValue="xxxxx" \
ParameterKey=RedHatSubscriptionPoolID,ParameterValue="xxxxx" \
ParameterKey=idContrailCommandPublicSubnet1,ParameterValue="subnet-00e02af1c495cc3ea" \
ParameterKey=idControllersPrivateSubnet1,ParameterValue="subnet-06f4f9b86b2fcfebe" \
ParameterKey=idControllersPrivateSubnet2,ParameterValue="subnet-07ac24ae57e7c47b3" \
ParameterKey=idControllersPrivateSubnet3,ParameterValue="subnet-01ee100098dc4fe1b" \
ParameterKey=idWorkersPrivateSubnet1,ParameterValue="subnet-06f4f9b86b2fcfebe" \
ParameterKey=idWorkersPrivateSubnet2,ParameterValue="subnet-07ac24ae57e7c47b3" \
ParameterKey=idWorkersPrivateSubnet3,ParameterValue="subnet-01ee100098dc4fe1b" \
ParameterKey=idVPCCONTROLLERS,ParameterValue="vpc-008ead9e50c0a8a7f" \
ParameterKey=idVPCWORKERS,ParameterValue="vpc-008ead9e50c0a8a7f" \
ParameterKey=SGSubnet2,ParameterValue="10.20.30.0/24" \
ParameterKey=SGSubnet3,ParameterValue="10.20.31.0/24" \
ParameterKey=SGSubnet4,ParameterValue="10.20.32.0/24" \
ParameterKey=PodIPAM,ParameterValue="10.20.35.0/24" \
ParameterKey=ServiceIPAM,ParameterValue="10.20.36.0/24"
---------------------------
```

Note: All fields, can be changed, as can the region. However for the first run we suggest you go with the defaults.
Note: For production deployments you should change the flavour from m4.2xlarge to m4.4xlarge

1. Get a coffee, it typically takes 6 minutes for the CloudFormation stack to build, then another 45 minutes for the Openshift Ansible deployer to complete.
2. Once the stack completes the CloudFormation outputs tab will show you how to access the hosts and Contrail Command UI.
3. To watch the deployment, ssh to the Infra node, "tail -f /var/log/bootstrap.log"

Post Installation Steps:

Once the deployer has completed:

Setup the admin password;

1. ssh to the master (see the stack outputs tab)
2. sudo bash
3. htpasswd -b /etc/origin/master/htpasswd admin contrail123
4. oc adm policy add-cluster-role-to-user cluster-admin admin
5. oc login -u admin

To access the Contrail UI over internet, use the link in outputs. Default login is admin contrail123.

To access the OpenShift UI over internet, use the Link provided in the stack outputs. As we use private hostnames in the stack, unless you have the opportunity to add public DNS entries, on your Mac/pc you should add the following entry to /etc/hosts.

/etc/hosts
[public ip of contrail command] [fqdn of a master node]

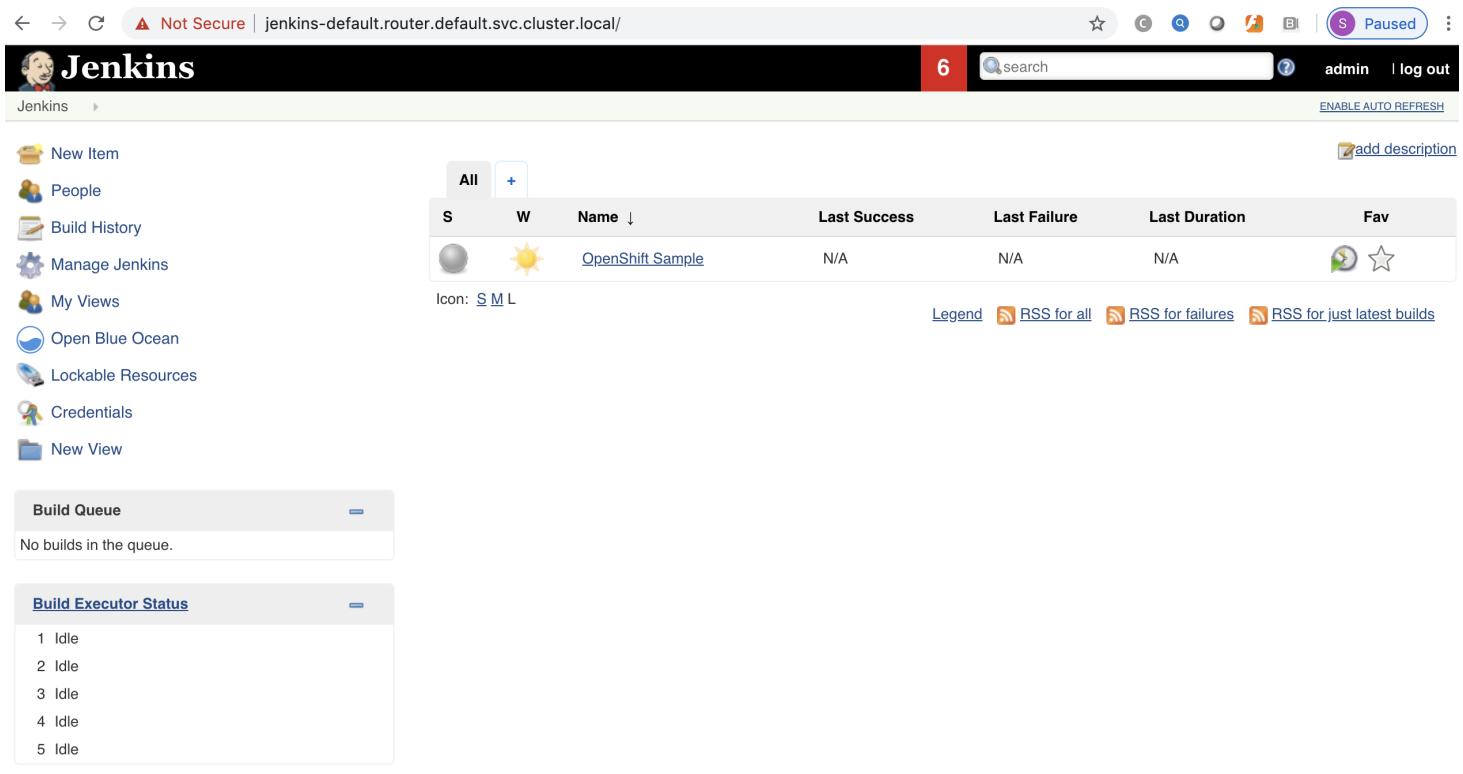
for example, my /etc/hosts
3.126.125.31 ip-100-72-100-16.eu-central-1.compute.internal
3.126.125.31 ip-100-72-100-17.eu-central-1.compute.internal
3.126.125.31 ip-100-72-100-18.eu-central-1.compute.internal
3.126.125.31 jenkins-default.router.default.svc.cluster.local

3.126.125.31 alertmanager-main-openshift-monitoring.router.default.svc.cluster.local
3.126.125.31 grafana-openshift-monitoring.router.default.svc.cluster.local
3.126.125.31 prometheus-k8s-openshift-monitoring.router.default.svc.cluster.local

Note: The last three dns entries are for the monitoring platform endpoints, these come from "oc -n openshift-monitoring get routes" and with https:// can be used to reach the integrated prometheus/grafana monitoring platform.

Running a quick test OpenShift application:

1. Open the Openshift UI (see stack outputs for the link)
2. Select the Default project (top right)
3. Select the Catalog
4. Select the Jenkins(Ephemeral) app. Deploy it.
5. Click overview (top left) You will now see an http link for your new app at the top. For example. http://jenkins-default.router.default.svc.cluster.local
6. Unless you setup a real DNS domain and fqdn for your apps (see stack parameter OpenShiftAppDNSSubDomain). Add this FQDN to your Mac/pc /etc/hosts (see above example), again point to the public address of Contrail Command.
7. Wait 4 minutes for the app to come up.
8. Point your browser to the http shown in the UI and you will see your web app.
9. This also works for https based apps such as Jenkins.

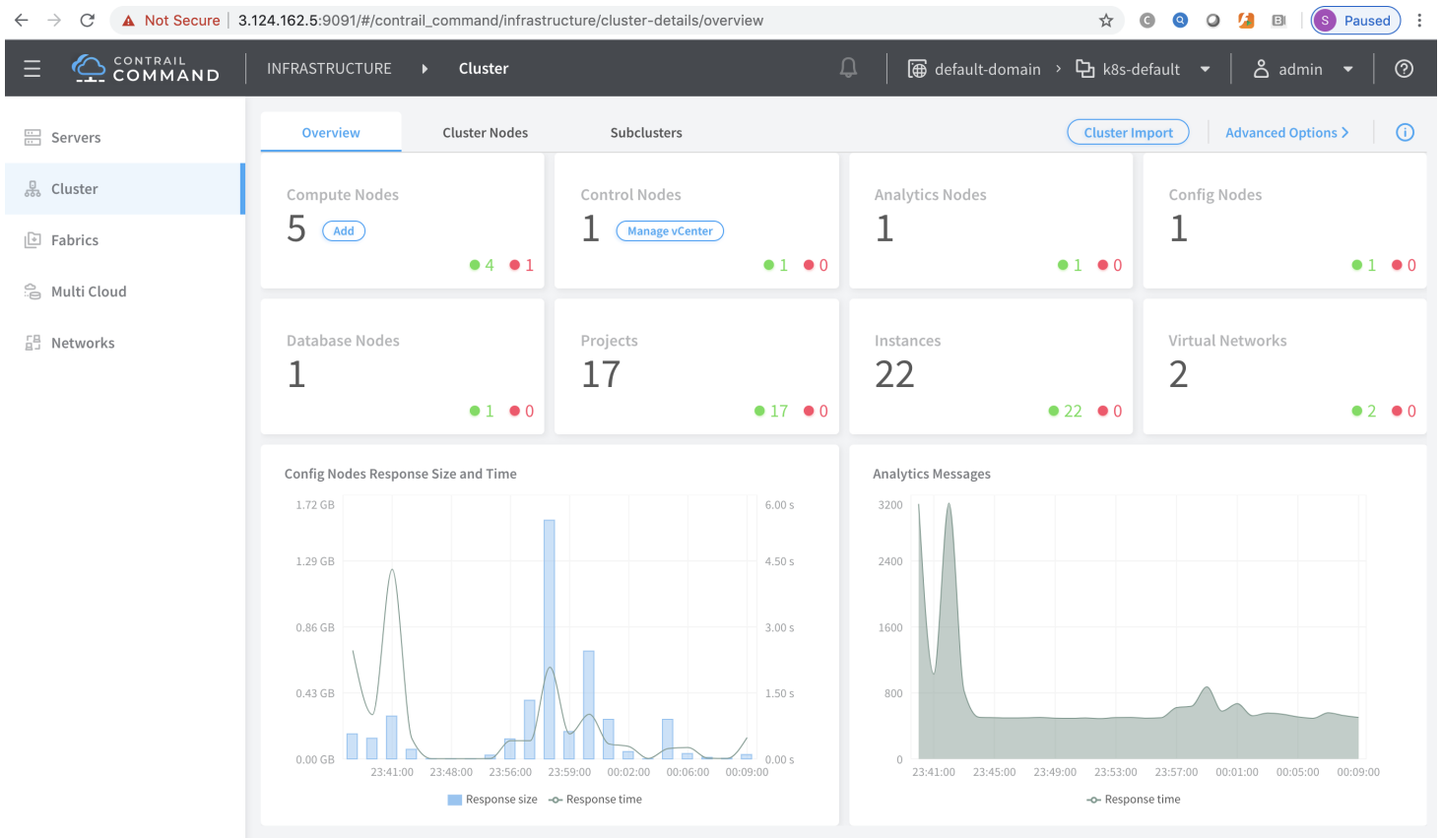


What does this AWS stack build:

1. AWS infrastructure across three availability zones, including OAM access over internet
2. Contrail Command
3. A Contrail command server which is built and configured.
4. An OpenShift infra server, used to deploy the Ansible and to host Contrail and OpenShift infra services.
5. An OpenShift Master server which is built and configured.
6. Three OpenShift nodes (workers) in three AZs, build and configured.

Importing your OpenShift cluster into Contrail Command

1) ssh into the Contrail Command EC2 instance (use the stack outputs for the link)
Example: ssh -i [your ContrailKey1 private key file] centos@52.28.213.130
2) get the ose-install file (change to ip in the example below to your infra1 node ip)
sudo bash
cat ~/.ssh/id_rsa.pub | sshpass -p 'EfrtGF5EDF_d54ERrf' ssh -o StrictHostKeyChecking=no root@100.72.100.11 'cat >> .ssh/authorized_keys'
scp root@100.72.100.11:/root/openshift-ansible/inventory/ose-install /tmp/ose-install
3) delete the existing contrail command cluster
docker stop $(docker ps -aq)
docker rm $(docker ps -aq)
4) deploy a new cluster importing the openshift ose file at the same time
docker run -td --net host -e orchestrator=openshift -e action=import_cluster -v /tmp/command_servers.yml:/command_servers.yml -v /tmp/ose-install:/instances.yml --privileged --name contrail_command_deployer hub.juniper.net/contrail/contrail-command-deployer:1912.32-rhel
5) Login
Login to the Contrail Command UI after a few minutes (see stack outputs)



Adding a public network and floating ip pool

in the stack, as we build the use-install file we add this line in order to tell the Contrail CNI where to assign the floating IP's.
public_fip_pool="{'domain': 'default-domain', 'project': 'k8s-default', 'network': 'k8s-public', 'name': 'default'}

To add the network into contrail follow these steps
1)open the contrail UI
2)config->networking->networks project=[default-project but cannot add + (add)]
name=k8s-public
subnet user defined
CIDR=[public subnet] example:172.30.195.240/24
advanced
shared=tick
Extend to physical router [select your MX gateway if you have one]
Router Targets [set your public network ASN and Target if you have them]
3) floating ip pools + (add)
network=k8s-public
name=default