**Contrail-installer**

Contrail-installer is a set of scripts and utilities to quickly build, install, configure and deploy OpenContrail. It can be used with pre-build packages (e.g. Launchpad PPA) or from sources. It is typically used in conjunction with devstack.

**Versions**

The contrail-installer master branch generally points to trunk versions of OpenContrail components whether sources or snapshots. For older, stable versions, use appropriate release name.

Currently contrail-installer supports the following (support the latest version of devstack, as of now it’s mitaka):

contrail-installer: sources master

contrail-installer: sources R3.0

contrail-installer: packages R2.20

**Topology Supported**

Currently contrail-installer support the following topology:

1. All in one or Single node setup where all the contrail roles (config, control, compute, webui, analytics, database and devstack) are configured on single node/server.
2. Controller and multiple compute. In this topology a single controller (with all contrail roles) and multiple compute on different node/server.

**Contrail-installer repo:**

Devstack generally uses stack user (non root) for installation of devstack and same is maintained for installation of opencontrail also using contrail-installer.

Create new user “stack” and give sudo permissions in /etc/sudoers file as shown below:

root ALL=(ALL:ALL) ALL

stack ALL=(ALL:ALL) ALL

The contrail-installer repo can be checked out from github.

Clone the contrail-installer to the node

git clone [git@github.com:Juniper/contrail-installer](mailto:git@github.com:Juniper/contrail-installer)

OpenContrail uses localrc to contain all local configuration and customizations. Copy the localrc-all file as shown below:

cd contrail-installer

cp samples/localrc-all localrc

Below are the important keys to be set in the localrc file

CONTRAIL\_DEFAULT\_INSTALL - Set this to True for installation from OpenContrail binary packages. When set to False, source from trunk OpenContrail will be downloaded and compiled.

LAUNCHPAD\_BRANCH=PPA - Applicable only when CONTRAIL\_DEFAULT\_INSTALL is set to True. It specifies to use released binary packages for installation. Default is to use latest snapshots as this knob is commented out by default in sample localrc.

PHYSICAL\_INTERFACE - This is external interface Vrouter should bind to. It should have a valid IP address configured. For example eth0

INSTALL\_PROFILE - Set this to ALL to for an all in one node. You can specify it to a particular role e.g. COMPUTE to make the node act only as compute node.

USE\_SCREEN - Set this to True to launch contrail modules in a screen session called "contrail". Connect to screen session for any troubleshooting of contrail modules.

LOGFILE - Specify logfile for contrail.sh runs. By default this is log/contrail.log in contrail-installer directory

**Launching Opencontril+Devstack (All-in-one Node)**

This can be done in 2 ways:

1. Staring Opencontrail and Devstack script separately one after other manually
2. Using task.sh (under utilities folder in contrail-installer) to install Opencontrail and Devstack automatically and execute basic sanity testcases

**Manual Method to install opencontrail and Devstack:**

OpenContrail uses localrc to contain all local configuration and customizations. Copy the localrc-all file after cloning the contrail-installer from github as shown below:

cd contrail-installer

cp samples/localrc-all localrc

Below are the important keys to be set in the localrc file:

CONTRAIL\_DEFAULT\_INSTALL - Set this to True for installation from OpenContrail binary packages. When set to False, source from trunk OpenContrail will be downloaded and compiled. When this flag is set to False, add the following entries in sources.list for installation of packages required for installation.

deb http://ppa.launchpad.net/opencontrail/ppa/ubuntu trusty main

deb-src http://ppa.launchpad.net/opencontrail/ppa/ubuntu trusty main

LAUNCHPAD\_BRANCH=PPA - Applicable only when CONTRAIL\_DEFAULT\_INSTALL is set to True. It specifies to use released binary packages for installation. Default is to use latest snapshots as this knob is commented out by default in sample localrc.

PHYSICAL\_INTERFACE - This is external interface Vrouter should bind to. It should have a valid IP address configured. For example eth0

INSTALL\_PROFILE - Set this to ALL to for an all in one node. You can specify it to a particular role e.g. COMPUTE to make the node act only as compute node.

USE\_SCREEN - Set this to True to launch contrail modules in a screen session called "contrail". Connect to screen session for any troubleshooting of contrail modules.

LOGFILE - Specify logfile for contrail.sh runs. By default this is log/contrail.log in contrail-installer directory

Contrail-installer uses contrail.sh to install/launch opencontrail and supports following options:

**build** ... to build OpenContrail – Download the dependencies, source code and builds the opencontrail system

**Install** ... to Install OpenContrail – Installs the opencontrail system by using the binaries (from generated build or from the PPA location) and required configuration files.

**configure** ... to Configure & Provision – overrides the environment values from localrc and replaces it with the generated configuration files.

**start** ... to Start OpenContrail Modules/daemons

**stop** ... to Stop OpenContrail Modules

**clean** ... to cleanup the database.

**restart** ... to Restart OpenContrail Modules without resetting data

**Launching OpenContrail**:

Follow the steps below to configure/install opencontrail.

cd contrail-installer

cp samples/localrc-all localrc (edit localrc as needed)

./contrail.sh build

./contrail.sh install

./contrail.sh configure

./contrail.sh start

The success of each step is determined by exit status of the script, exit status of zero is success and non-zero exit is failure. Refer to the detailed log “contrail.log” under log folder in case of failure.

If keyword USE\_SCREEN is set to TRUE, contrail daemons will be launched in screen session by name “contrail”, use screen –x contrail to run through various tabs to see the contrail modules/daemon running.

**Launching devstack:**

Trunk of contrail-installer works with stable/mitaka, follow the steps below to launch devstack.

Checkout the devstack repo from:

git clone [git@github.com:openstack-dev/devstack](mailto:git@github.com:openstack-dev/devstack)

cd devstack

git checkout stable/mitaka

A glue file is needed in the interim till it is upstreamed to devstack

cp ~/contrail-installer/devstack/lib/neutron\_plugins/opencontrail lib/neutron\_plugins/

Use sample localrc:

cp ~/contrail-installer/devstack/samples/localrc-all localrc

edit the localrc file to update the physical interface info.

Run stack.sh

./stack.sh

**Restarting OpenContrail+Devstack**

If you need to restart OpenContrail or Devstack for some reason, currently they need to be synchronized. So

cd ~/devstack

./unstack.sh

cd ~/contrail-installer

./contrail.sh restart

cd ~/devstack

./stack.sh

if issues persist, it might be helpful to reboot server or VM and repeat the steps below

cd ~/contrail-installer

./contrail.sh start

cd ~/devstack

./stack.sh

Verify installation

1) screen -x contrail and run through various tabs to see various contrail modules are running

2) Run utilities/contrail-status to see if all services are running (need to set CONTRAIL\_DIR and DEVSTACK\_DIR to contrail-installer and devstack folder respectively before running the command)

Running sanity

Note that default sample localrc enables simple gateway. A script is available that will create a virtual network, launch two VMs, ping each VM from host and then SSH into it. Follow the steps below:

cd ~/contrail-installer/utilities

export CONTRAIL\_DIR=~/contrail-installer

export DEVSTACK\_DIR=~/devstack

./contrail-sanity

**Second approach**

**Automating contrail.sh and devstack (all-in-one Node)**

contrail-installer/utilities/task.sh attempts to automate steps required by sequential runs of contrail.sh and devstack. It works off a configuration file, default called auto.conf is provided. Below are the important keywords to check:

CONTRAIL\_INSTALLER\_BRANCH – The branch of contrail-installer to be cloned. Default being MASTER.

ENABLE\_BINARY – Set to TRUE for Binary installation and False for source code installation. If set to False, add the following entries in sources.list for installation of packages required for installation.

deb http://ppa.launchpad.net/opencontrail/ppa/ubuntu trusty main

deb-src http://ppa.launchpad.net/opencontrail/ppa/ubuntu trusty main

LAUNCHPAD\_BRANCH – If ENABLE\_BINARY is set to True, uncomment and set this to appropriate packages. Default being sanapshots or else this can be uncommented.

DEVSTACK\_CLONE\_BRANCH - branch of devstack to be cloned ex: stable/mitaka

Following example launches task.sh in binary PPA mode while using R2.20 packages. See auto.conf for more options to launch in source mode or with use of snapshots

$ cd ~/contrail-installer/utilities

$ diff auto.conf my.conf

17c17

< ENABLE\_BINARY=False

---

> ENABLE\_BINARY=True

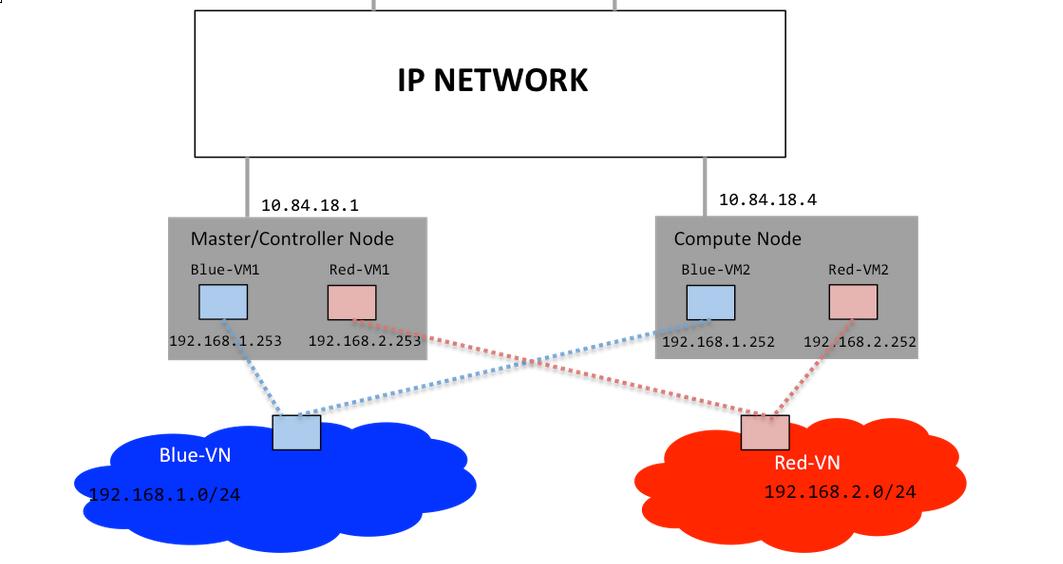
22a23

> LAUNCHPAD\_BRANCH=r2.20

$ ./task.sh my.conf

**Multi Compute Node Setup**

This model of deployment contains multiple nodes with all configured in one node and only compute services in other nodes. I.e. one physical host or VM running controller, compute services together and one physical hosts or VMs running only compute services.



**Launching opencontrail on controller Node.**

Follow the steps to to launch opencontrail and devstack manually on controller node:

1. Download contrail-installer on opencontrail installer node from github.

git clone git@github.com:Juniper/contrail-installer

1. Copy the localrc file and update it with required keywords

cd contrail-installer

cp samples/localrc-all localrc

CONTRAIL\_DEFAULT\_INSTALL - Set this to True for installation from OpenContrail binary packages. When set to False, source from trunk OpenContrail will be downloaded and compiled. When this flag is set to False, add the following entries in sources.list for installation of packages required for installation.

deb http://ppa.launchpad.net/opencontrail/ppa/ubuntu trusty main

deb-src http://ppa.launchpad.net/opencontrail/ppa/ubuntu trusty main

LAUNCHPAD\_BRANCH=PPA - Applicable only when CONTRAIL\_DEFAULT\_INSTALL is set to True. It specifies to use released binary packages for installation. Default is to use latest snapshots as this knob is commented out by default in sample localrc.

PHYSICAL\_INTERFACE - This is external interface Vrouter should bind to. It should have a valid IP address configured. For example eth0

INSTALL\_PROFILE - Set this to ALL to for an all in one node. You can specify it to a particular role e.g. COMPUTE to make the node act only as compute node.

USE\_SCREEN - Set this to True to launch contrail modules in a screen session called "contrail". Connect to screen session for any troubleshooting of contrail modules.

LOGFILE - Specify logfile for contrail.sh runs. By default this is log/contrail.log in contrail-installer directory

1. Follow the below steps to launch opencontrail:

contrail.sh build

contrail.sh install

contrail.sh configure

contrail.sh start

1. Download devstack from github and checkout the stable supported devstack release.

git clone –b stable/mitaka https://github.com/openstack-dev/devstack.git

1. Copy the localrc file and update the physical interface info and copy the contrail neutron plugins to devstack neutron plugin folder of devstack.

cd devstack

cp samples/localrc-all localrc

Update localrc with correct PHYSICAL\_INTERFACE value

1. A glue file is needed in the interim till it is upstreamed to devstack

cp ~/contrail-installer/devstack/lib/neutron\_plugins/opencontrail lib/neutron\_plugins/

1. Run stack.sh

Follow the steps below to launch COMPUTE role on other physical nodes or VMs:

1. Download contrail-installer on opencontrail installer node from github.

git clone git@github.com:Juniper/contrail-installer

1. Edit localrc as shown below:

cd contrail-installer

cp samples/localrc-all localrc

INSTALL\_PROFILE =COMPUTE

SERVICE\_HOST=<controller -ip>

CONTROL\_IP=<controller-ip>

1. Follow the below steps to launch opencontrail:

contrail.sh build

contrail.sh install

contrail.sh configure

contrail.sh start

1. Download devstack from github and checkout the supported devstack release

git clone –b stable/mitaka https://github.com/openstack-dev/devstack.git

1. Create localrc file with the following keywords

cd devstack

# change this to master/controller node's ip

SERVICE\_HOST=<controller-ip># control1

# the interface that contrail's vhost0 should take over PHYSICAL\_INTERFACE=eth0

Q\_PLUGIN=opencontrail STACK\_DIR=$(cd $(dirname $0) && pwd)

# log all screen output to this directory SCREEN\_LOGDIR=$STACK\_DIR/log/screens LOG=True

DEBUG=True

LOGFILE=$STACK\_DIR/log/stack.log

LOGDAYS=1

ENABLED\_SERVICES=n-cpu,rabbit,g-api,neutron,n-novnc,n-xvnc Q\_USE\_DEBUG\_COMMAND=True

PHYSICAL\_NETWORK=default

MYSQL\_HOST=$SERVICE\_HOST

RABBIT\_HOST=$SERVICE\_HOST

Q\_HOST=$SERVICE\_HOST

GLANCE\_HOSTPORT=$SERVICE\_HOST:9292 DATABASE\_PASSWORD=contrail123

RABBIT\_PASSWORD=contrail123

SERVICE\_TOKEN=contrail123

SERVICE\_PASSWORD=contrail123

ADMIN\_PASSWORD=contrail123

DATABASE\_TYPE=mysql

# repo proto is https or (default) ssh. Leave commented for ssh # CONTRAIL\_REPO\_PROTO=https

# proto for openstack bits. Use HTTPS if git is firewalled GIT\_BASE=https://git.openstack.org

# use contrail VIF driver with NOVA NOVA\_VIF\_DRIVER=nova\_contrail\_vif.contrailvif.VRouterVIFDriver

1. cp ~/contrail-installer/devstack/lib/neutron\_plugins/opencontrail lib/neutron\_plugins/
2. run stack.sh
3. Verify if the compute nodes have comeup by
4. running contrail-status command from contrail-installer/utilities folder
5. login to contrail-webui and check if the dashboard for the number of instances of compute nodes.
6. Run sanity from controller node to verify the TCs

cd ~/contrail-installer/utilities

export CONTRAIL\_DIR=~/contrail-installer

export DEVSTACK\_DIR=~/devstack

./contrail-sanity