Jaspreet Singh 201202078
Shikher Somal 201201091
Ruchir Sharma 201405558
Priyanka Bajaj 201405540

Openstack with Docker - Documentation

Team 17

Introduction

The project deals with building a three-tier application for using Openstack and Docker. This report will act as a documentation for the project. Openstack is used to launch virtual machines and resource instances such as nova compute servers, network servers given a flavor and image. Heat is an orchestration which enables us to launch multiple cloud applications at once. We will use heat to launch docker containers which can communicate among themselves using the Docker plugin for heat. Given Below is a step by step procedure on how to achieve this task.

Note: All the steps require the proxy variables set in your bashrc file present in the user directory. For this, if you are behind a proxy, add these lines in your .bashrc file. Here yourip is the ip assigned to you on the network. You can find this by ifconfig

```
no_proxy=localhost,127.0.0.0/8,yourip

http_proxy=http://proxy.iiit.ac.in:8080/

https_proxy=https://proxy.iiit.ac.in:8080/

NO_PROXY=localhost,127.0.0.0/8,yourip

HTTP_PROXY=http://proxy.iiit.ac.in:8080/

HTTPS_PROXY=https://proxy.iiit.ac.in:8080/
```

Installing Openstack

We used Ubuntu 14.04 to install the Openstack

1. To start off you need git
   sudo apt-get -y install git
2. Download latest source code of DevStack
   git clone https://github.com/openstack-dev/devstack.git
   cd devstack
   git checkout stable/juno
   You can also get the kilo or liberty release. But we have used juno for our project
3. Make sure you have set the $no_proxy, $http_proxy and $https_proxy variables:
   echo $no_proxy
   localhost,127.0.0.0/8,10.1.34.125
   echo $http_proxy
   http://proxy.iiit.ac.in:8080/
   echo $https_proxy
   https://proxy.iiit.ac.in:8080/
4. Make sure you have heat in the enabled services in your stackrc, if not add the following line in the ENABLED_SERVICES:
   ENABLED_SERVICES+=,h-eng,h-api,h-api,cfn,h-api-cw
5. Run the stack command
   ./stack.sh
   This will take around 15 minutes in a 4GB RAM system. You will see that ./stack.sh completed in XXXX seconds if the command exited normally. Usually if this exits with error, then you'll have to reinstall the required package. In our case, this was for virtualenv, tox, six. You can do this by
   sudo -EH pip uninstall <package>
   and
   sudo -EH pip install <package>

6. If you want to take down OpenStack for some time or want to turn off machine, run unstack.sh. To get it up and running again, run rejoin-stack.sh.

7. When we used to do rejoin-stack.sh, the Keystone service would not start automatically and so I could not login to OpenStack dashboard. To fix this I had to manually start it. To start the Keystone service manually, run following command. /opt/stack/keystone/bin/keystone-all –config-file /etc/keystone/keystone.conf -d &

8. The ./rejoin-stack.sh command will open a screen which is many terminals under a terminal . This screen runs different services individually such as h-api,n-api et. To scroll between the screen, press Crtl+a then p for previous screen and Crtl+a and n for next screen. For all screens run Crtl+a and Shift+' .For scolling up in a screen press Crtl+a and Crtl+[.

Installing Docker

The [docker documentation/ installation] will give you the steps to get docker installed in your system. Following are the steps for a linux system

1. Update repository
   sudo apt-get update

2. Install docker
   sudo apt-get install docker-engine

3. To add the Docker repository key to your local keychain, run following commands
   sudo apt-key adv –keyserver hkp://keyserver.ubuntu.com:80 –recv-keys 36A1D7869245C8950F966E92D8576A8BA88D21E9

4. To add the Docker repository to your apt sources list, update and install the lxc-docker package, run following commands
   sudo sh -c "echo deb https://get.docker.com/ubuntu docker main > /etc/apt/sources.list.d/docker.list"
   sudo apt-get update
   sudo apt-get install lxc-docker

5. To verify everything was running fine, run the following command-
   sudo docker run hello-world
   This will give you something starting with
   Hello from Docker.

6. You could also run the ubuntu image on docker by running
   sudo docker run -i -t ubuntu /bin/bash
   This should pull ubuntu image from Docker respository, run it and open the terminal.

7. If you are behind a proxy. Uncomment the line starting with DOCKER_OPTS to enable a public dns server and export proxy variables
   export http_proxy="http://proxy.iiit.ac.in:8080"
   export https_proxy="https://proxy.iiit.ac.in:8080"

Docker Plugin for Heat (DockerInc:: Docker::Container)

Once you have OpenStack and Docker installed. You can launch containers instead of VMs through Heat, the orchestration engine of OpenStack, with the help of Heat-Docker plugin. To install the plugin follow below steps.

1. Download the heat directory.

2. Delete the tests folder under the heat/contrib/heat_docker/heat_docker directory.

3. In the top-level heat_docker folder, run pip install -r requirements.txt. Note that you might need to do an apt-get install python-pip first. This will install the docker-py Python module, which is required by the Docker plugin.

4. Copy the heat/contrib/heat_docker/heat_docker/resources/docker_container.py to /var/lib/heat/docker folder

5. Modify heat configuration file, /etc/heat/heat.conf to include the above full path, /var/lib/heat/docker to the plugin in the plugin_dirs

6. Restart the heat engine. To do it, if you have a service for heat-engine you can restart it. Otherwise you need to find a command to restart the heat-engine or if u don't mind getting OpenStack down, simply run unstack.sh and then rejoin-stack.sh.

7. To verify the plugin was integrated successfully, run the command heat resource-type-list. You should see DockerInc::Docker::Container listed in the results. You may need to source openrc before running the above command.

8. Now you may face issues after you execute run pip install -r requirements.txt and then try to rejoin-stack. There might be versioning problem with Six python module. To resolve this, run following commands
   pip install -U six
   sudo easy_install pip

Use Heat Orchestration Template (HOT) to launch containers

Once you have Docker and Heat-Docker plugin installed, you can launch containers in OpenStack environment.
For this you need to use a Heat template. In this you specify resources to be launched, their types and properties etc. To know more about these templates visit HOT Guide.
In case of containers, the type of a resource will be DockerInc::Docker::Container, which depends on the plugin you use and is listed in result of heat resource-type-list command.
In properties section, we need two fields.
image - whose value will be the image of the container you want to launch. Note that this image should have already be present locally. It is not pulled automatically when we try to launch it through Heat.
docker_endpoint - this is the address at which docker is listening. By default when we install Docker it is not listening, we need make Docker service listen to some address. For running docker daemon on a port, use command sudo <path to>/docker -H 0.0.0.0:<port> -d &. Now we can specify this address as docker_endpoint in HOT.

Our github repository gives few basic heat templates with one having described above.

To test this, login to OpenStack dashboard. Go to Project->Orchestration->Stacks->Create New Stack/Launch Stack. Here provide the above template and launch.

Make Containers Talk

Now we want to make a three tier application in which each tier is a container itself. These containers, in our case php,mysql and apache container for three tier or a mysql and a wordpress container for a two tier application. The explanation of the tiers go as follows:

In order to communicate with the DB, blog tier needs IP address of DB container. We can provide this IP address to blog container by passing this IP address as environment variable to blog container. For this use env field in properties section. We can get IP address of a container with ‘get_attr: [<resource_name>, network_ip]’.

To test this

1. Pull two docker images rishineo/wordpress_file2 and rishineo/mysql_dbpass.
2. Login to OpenStack dashboard. Go to Project->Orchestration->Stacks->Create New Stack/Launch Stack and provide the above template. Launch.
3. Enter into the DB container from command line. Create a database 'wordpress’. (You may exit the container now.)
4. Visit the IP address of the blog in browser. Note that IP can be found in Stack details. You will see a WordPress page.

This can be similarly done for a three tier application. There will now be three containers with three docker end points and each one of them exposing their individual ports to the others.

The php container will depend on database to receive dbname and dbpass. The apache server will expose the ports used by php frontend to act as a web server.

More On The Project

Openstack has apparently stopped supporting the Docker plugin for Heat. This has led to a new product - Magnum, a standalone project in Openstack to facilitate the launching of containers using bay. But it works with kubernetes which was beyond the scope of our project but definitely a possible extension.