Building BigTop using Docker container

Introduction

Build

Sources
- Upstream: https://git.linaro.org/leg/bigdata/bigtop-trunk.git/
- Root upstream: https://github.com/apache/bigtop
- Release-1.2.0 is used in this Wiki

Setup Environment
- Ubuntu 16.04 64bit for AArch64, or CentOS7 64bit for AArch64
- Oracle JDK 1.8.0_111-b14

Pre-Requisites
Oracle JDK 1.8.0_111-b14

Dependencies
maven@v3.0.5, nodejs@v4.2.6, npm@3.5.2, brunch@1.7.10, phantomjs@1.9.8, python>=2.6, python-dev, rpm, yum, g++

Build Steps

Install Pre-requisites

For Ubuntu:

apt install docker.io ruby git unzip openjdk-8-jre

sudo apt-get install docker-engine

For Centos:

sudo docker info
sudo docker version
service docker status
service docker restart
docker run hello-world
groupadd docker

git clone https://github.com/odpi/bigtop.git

cd bigtop
./gradlew docker-provisioner-destroy
## ./gradlew -Pconfig=vagrantconfig.yaml -Pnum_instances=3 -Prun_smoke_tests=true docker-provisioner
## (cd /tmp/odpi/bigtop-deploy/vm/vagrant-puppet-docker & & vagrant
## ssh bigtop1)

1. Prepare the Docker build environment
Using pre-built Docker images

The Dockerfiles to regenerate them are located in ./docker/bigtop-slaves.

#docker pull bigtop/slaves:1.1.0-ubuntu-16.04-aarch64

Start a container with the latest image:

docker run --rm -v `pwd`:/ws bigtop/slaves:1.1.0-ubuntu-16.04-aarch64 /bin/bash

sudo docker run -u jenkins --rm -v `pwd`:/ws$DOCKER_SELINUX --workdir /ws bigtop/slaves:trunk-ubuntu-16.04 bash -l -c './etc/profile.d/bigtop.sh; . /gradlew allclean all'

The Job Bigtop-1.1.0 to create a repository is essentially:

#!/bin/bash -ex
# Workaround for wrong distributions (sic! ppc64el !) BIGTOP-2344
sed -i 's/Architectures: i386 amd64 source/Architectures: ppc64el source/' packages.gradle

docker run --rm -v `pwd`:/ws bigtop/slaves:1.1.0-$BUILD_ENVIRONMENTS bash -c '
A=rpm ; B=yum;
type -p dpkg && A=deb ;
type -p dpkg && B=apt ;
./gradlew allclean $A $B'

Now you are inside your Bigtop Docker environment and you can start building Hadoop packages.

2. Build BigTop RPMs

To build an RPM for a single project use ./gradlew <project name>-rpm. For example to build Spark RPM do:

# Build Spark RPM
cd ~/bigtop
./gradlew spark-rpm

To build all BigTop RPMs use ./gradlew rpm:

# Build all RPMs
cd ~/bigtop
./gradlew rpm

To list all gradle tasks run cd ~/bigtop & & ./gradlew tasks.
Then run ./gradlew spark-rpm.

Jenkins job
3. Download the RPMs

All generated RPMs are stored into `/home/bigtop/bigtop/build/**/rpm/RPMS/**/*.rpm` folders on the docker container.

```bash
$ ls -lah /home/bigtop/bigtop/build/**/rpm/RPMS/**/*.rpm
-rw-rw-r-- 1 bigtop bigtop 26M May 14 09:40 /home/bigtop/bigtop/build/flume/RPMS/noarch/flume-1.5.2-1.el6.noarch.rpm
-rw-rw-r-- 1 bigtop bigtop  6.2K May 14 09:40 /home/bigtop/bigtop/build/flume/rpm/RPMS/noarch/flume-agent-1.5.2-1.el6.noarch.rpm
```

Copy them to the shared `/rpm` volume to make them accessible by the Docker Host:

```bash
sudo cp /home/bigtop/bigtop/build/**/rpm/RPMS/**/*.rpm /rpm
```

Now you can copy them from the Docker Host into a local folder (default boot2docker password: tcuser):

```bash
scp -rp docker@<Docker Host IP>:/rpm/*.rpm <Your Local Folder>
```

Note: Make sure you check the RPM dependencies before you install them. When you install the RPMs on your own Hadoop distro you may need to use the `rpm` with the `--nodeps` option to prevent dragging nonrequired dependencies.