Ansible Playbook - Setup Hadoop CDH5 Using `tarball`.

Setting up Hadoop using Ansible, we will be using cdh5 tarball for installation of the cluster.

Table of Contents

This is a simple Hadoop playbook, to quickly start hadoop running in a cluster.

Here is the Script Location on Github: https://github.com/zubayr/ansible_hadoop_tarball

Below are the steps to get started.

Get the script from Github.

Below is the command to clone.

```
ahmed@ahmed-server ~]$ git clone https://github.com/zubayr/ansible_hadoop_tarball
```

Before we start.

Download `hadoop-2.3.0-cdh5.1.2.tar.gz` to `file_archives` directory.

Download `jdk-7u75-linux-x64.tar.gz` to `file_archives` directory.

Details about each Playbook 'Roles'.

Details about each Role.

commons

This role is used to update OS parameters and will update the below files.

1. `sysctl.conf` Update swapiness, networking and more. Info in `defaults/main.yml`
2. `limits.conf` Update `soft` and `hard` limits.
3. `90-nproc.conf` Update user based limits and adding `hadoop_user` limits file.
4. `/etc/hosts` Update `hosts` file on the server - from `host_name` in `hosts` file.

`/etc/hosts` file will get the server information from the `[allnodes]` group in the `hosts` file.

NOTE : Commons will update the `HOSTNAME` of the server as well as per these entries.

jdk

This role install jdk1.7. Installation path - from `group_vars/all` with variable `java_home`.

ssh_known_hosts

This role will create ssh known hosts for all the hosts in the `hosts` file.

ssh_password_less

This role will make `hadoop_user` passwordless user for `hadoop` nodes.

cdh5_hadoop_commons_tarball

This role will install and configure hadoop installation. Update files.

1. `core-site.xml` Add Namenode.
2. `hdfs-site.xml` Update hdfs parameters - `default/main.yml`.
3. `mapred-site.xml` Update MR information.
4. `yarn-site.xml` Update Yarn.
5. `slaves` Update slaves information - `hosts` file.
6. `hadoop-env.sh` Update `JAVA_HOME` - `group_vars`.

post_install_setups

This is hadoop user creation after installation. If we need more users then we need to add them in role `post_install_setups`.

Current we will create a user called `stormadmin`. More details in `roles/post_install_setups/tasks/create_hadoop_user.y`

```
#
# Creating a Storm User on Namenode/ This will eventually be a edge node.
#
- hosts: namenodes
  remote_user: root
  roles:
    - post_install_setups
```

Step 1. Update below variables as per requirement.

Global Vars can be found in the location `group_vars/all`.

```
# ------------------------------------
# USERs
# ------------------------------------

hadoop_user: hdadmin
hadoop_group: hdadmin
hadoop_password: <encrypted_password_here_howto_below>

# Common Location information.
common:
  install_base_path: /usr/local
  soft_link_base_path: /opt
```

Step 2. User information come from `group_vars`.

Username can be changed in the Global Vars, `hadoop_user`. Currently the password is `hdadmin@123`

Password can be generated using the below python snippet.

```
# Password Generated using python command below.
python -c "from passlib.hash import sha512_crypt; \
            import getpass; print sha512_crypt.encrypt(getpass.getpass())"
```

Here is the execution. After entering the password you will get the encrypted password which can be used in the user creation.

```
ahmed@ahmed-server ~]$ python -c "from passlib.hash \
            import sha512_crypt; import getpass; print sha512_crypt.encrypt(getpass.getpass())"
Enter Password: *******
$6$rounds=40000$1qjG/2uPsmGK/2xnmOt80TjDwbof9rNvnYY6icCkdAR2qrFquirBtT1
ahmed@ahmed-server ~]$
```

Step 3. Update Host File.

IMPORTANT update contents of `hosts` file. In `hosts` file `host_name` is used to create the `/etc/hosts` file.

```
#
# All pre-prod nodes.
#
[allnodes]
10.10.18.30 host_name=ahmd-namenode
10.10.18.31 host_name=ahmd-datanode-01
10.10.18.32 host_name=ahmd-datanode-02
10.10.18.34 host_name=ahmd-resourcemanager
10.10.18.93 host_name=ahmd-secondary-namenode
10.10.18.94 host_name=ahmd-datanode-03
10.10.18.95 host_name=ahmd-datanode-04
```

```
#
# hadoop cluster
#

[namenodes]
10.10.18.30

[secondarynamenode]
10.10.18.93

[resourcemanager]
10.10.18.34

[jobhistoryserver]
10.10.18.34

[datanodes]
10.10.18.31
10.10.18.32
10.10.18.94
10.10.18.95

[hadoopcluster:children]
namenodes
secondarynamenode
resourcemanager
jobhistoryserver
datanodes

#
# sshknown hosts list.
#

[sshknownhosts:children]
hadoopcluster
```

Step 4. Executing yml.

Execute below command.

```
ansible-playbook ansible_hadoop.yml -i hosts --ask-pass
```