

# Ansible Playbook - Setup Zookeeper Using `tarball`.

This is a simple zookeeper playbook, to quickly start zookeeper running on a single or more nodes, in a clustered mode.

Here is the Script Location on Github: [https://github.com/zubayr/ansible\\_zookeeper\\_tarball](https://github.com/zubayr/ansible_zookeeper_tarball)

Below are the steps to get started.

## Before we start.

Download [zookeeper-3.4.5-cdh5.1.2.tar.gz](#) to `file_archives` directory.

Download [jdk-7u75-linux-x64.tar.gz](#) to `file_archives` directory.

## Get the script from Github.

Below is the command to clone.

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

## Step 1. Update below variables as per requirement.

Variables are located in `roles/zookeeper_install_tarball/default/main.yml`.

```
# Zookeeper Version.
zookeeper_version: zookeeper-3.4.5-cdh5.1.2

# Zookeeper Storage and Logging.
zookeeper_data_store: /data/ansible/zookeeper
zookeeper_logging: /data/ansible/zookeeper_logging
```

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

```
# -----
# USERS
# -----

zookeeper_user: zkadmin
zookeeper_group: zkadmin
zookeeper_password: <encrypted_password_here>

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

## Step 2. User information come from global\_vars.

Username can be changed in the Global Vars, `zookeeper_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$/VINUa2uPsmGK/2xnm0t80TjDwbof9rNvnYY6icCkdAR2qrFquirBtT1
ahmed@ahmed-server ~]$
```

## Step 3. Update playbook.

Update file called `ansible_zookeeper.yml` (if required) with `hosts` file in root of the directory structure. Below is the sample directory structure.

```
zookeeper.yml
hosts
global_vars
  --> all
file_archives
  --> zookeeper-3.4.5-cdh5.1.2.tar.gz
  --> ...
roles
  --> zookeeper_install_tarball
  --> ...
```

Below are the contents for `ansible_zookeeper.yml`

```
#
#-----
# ZOOKEEPER CLUSTER SETUP
#-----
#

- hosts: zookeepernodes
  remote_user: root
  roles:
    - zookeeper_install_tarball
```

Steps used in `zookeeper_install_tarball` role.

1. Create a user to running zookeeper service. **NOTE:** user information in `global_vars`.

2. Copy tgz file and extract in destination.
3. Changing permission to directory, setting `zookeeper_user` as the new owner.
4. Creating Symbolic link. **NOTE:** `soft_link_base_path` information in `global_vars`.
5. Updating Configuration File in Zookeeper.
6. Creating directory for Zookeeper.
7. Initializing `myid` file for Zookeeper.
8. Starting Zookeeper Service.

Here are the contents of `hosts` file. In `hosts` file `zookeeper_id` will be used to create an `id` in `myid` file, for each zookeeper running as a cluster.

```
#
# zookeeper cluster
#

[zookeepernodes]
10.10.18.25 zookeeper_id=1
10.10.18.87 zookeeper_id=2
10.10.18.90 zookeeper_id=3
```

#### Step 4. Executing yml.

Execute below command.

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