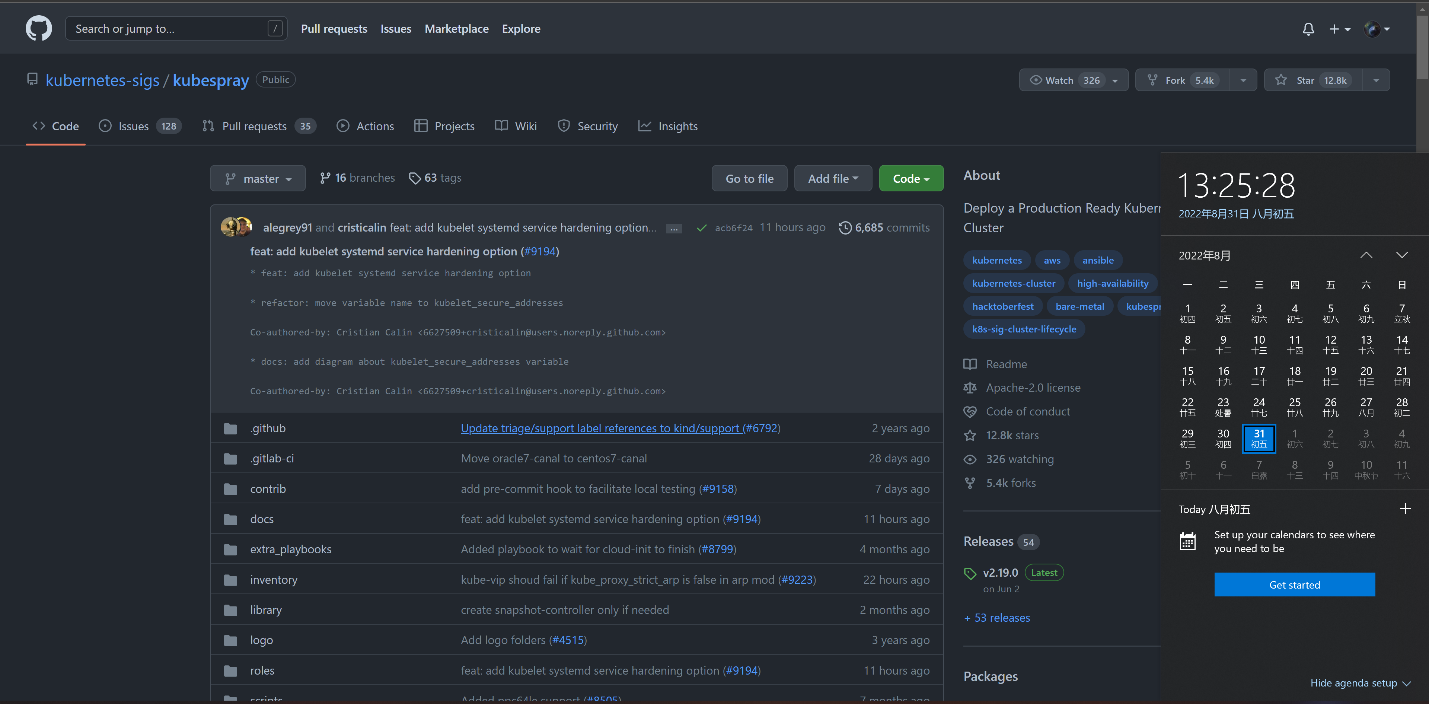
Using kubespray to setup k8s

***Attention: This document is only for internal environment and internal usage. Below is the version of kubespray supported:***

# What’s Kubespray

Kubespray is a composition of [Ansible](https://docs.ansible.com/) playbooks, [inventory](https://github.com/kubernetes-sigs/kubespray/blob/master/docs/ansible.md#inventory), provisioning tools, and domain knowledge for generic OS/Kubernetes clusters configuration management tasks.

Github: <https://github.com/kubernetes-sigs/kubespray>

k8s Documentation: <https://kubernetes.io/docs/setup/production-environment/tools/kubespray/>

# The environment you need to prepare to use kubespray

## Conda

1. wget <https://repo.anaconda.com/miniconda/Miniconda3-py39_4.12.0-Linux-x86_64.sh>
2. chmod +x ./Miniconda3-py39\_4.12.0-Linux-x86\_64.sh
3. ./Miniconda3-py39\_4.12.0-Linux-x86\_64.sh
4. source ~/.bashrc

## Create a virtual environment(easy to manage)

1. conda create --name env\_1 python=3.9.12 pip

// feel free to change env\_1 to any other name you want the environment to be called, but remember to be consistent below

1. conda activate env\_1
2. conda update -y -n env\_1 pip
3. pip install ansible==5.6.0 click
4. pip install netaddr
5. pip install ruamel.yaml

## Suggestion

follow the Quick Start given by <https://github.com/kubernetes-sigs/kubespray>

# What and where to config

## config ips of where to deploy k8s

As shown above, kubespray has scripts to help us config the IP where we would like to set up k8s.

These files below is where to be changed.

kubespray/inventory/mycluster/hosts.yaml

kubespray/inventory/mycluster/group\_vars/all/all.yml

kubespray/inventory/mycluster/group\_vars/k8s\_cluster/k8s-cluster.yml

## config ansible

Ansible has its own configurations, you may want to ajust them in different cases. Here are several links you can take a look.

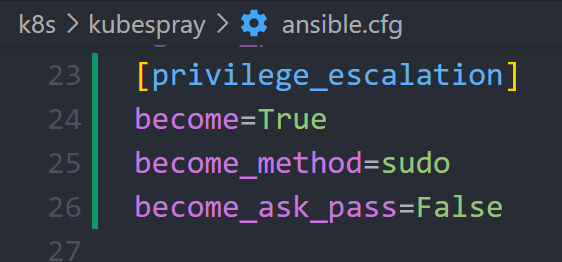
<https://docs.ansible.com/ansible/latest/reference_appendices/config.html>

<https://docs.ansible.com/ansible/latest/installation_guide/intro_configuration.html>

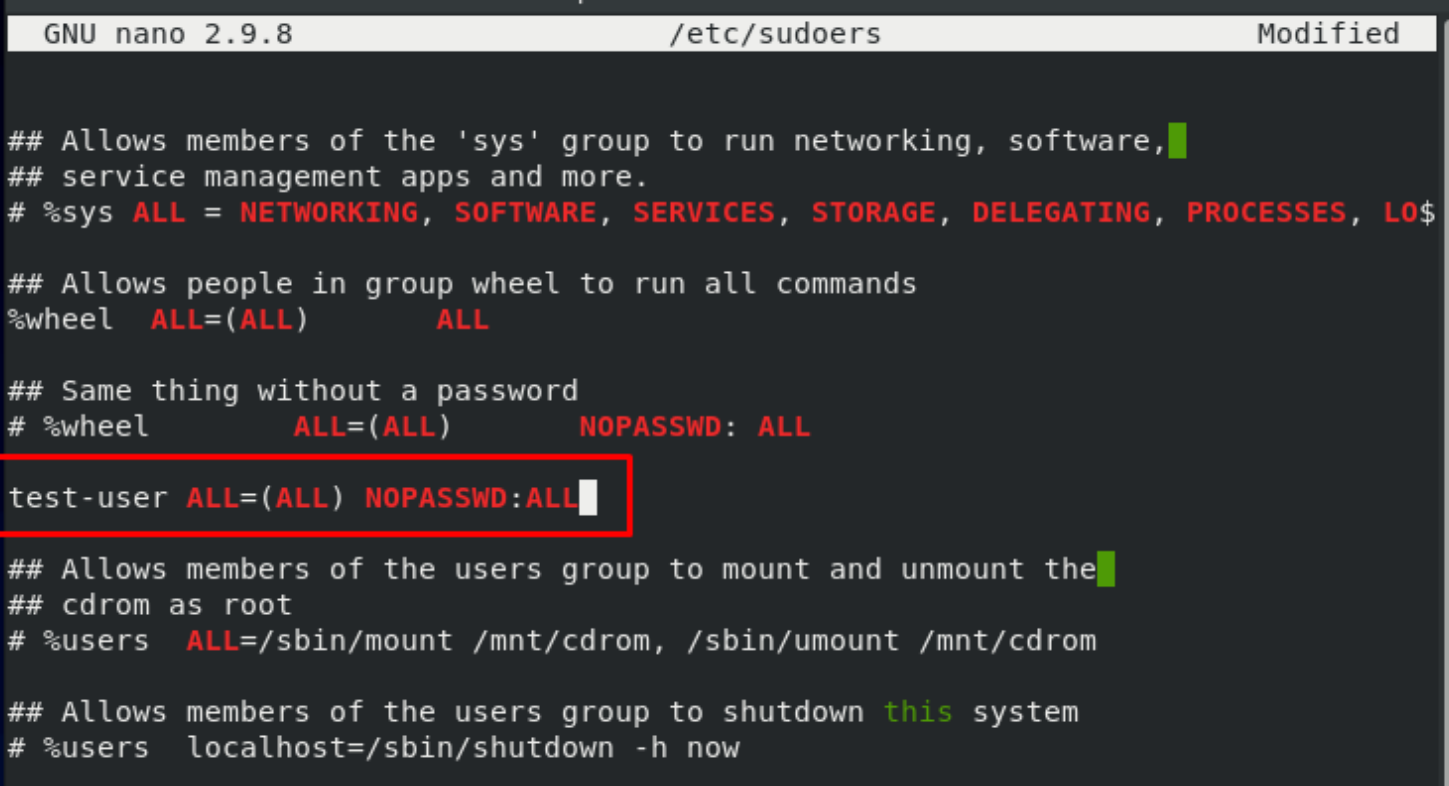
Cause the access to the node you want to deploy k8s on is necessary, you need to config how to connect to the specified node.

Ajust the file below and add these codes if don’t exist:

kubespray/ansible.cfg



Attention:

The key-value “become\_ask\_pass=False” need the user in “ssh user@IP” has the privilege to sudo without password. To give user this privilege, you can take a look of this link: <https://linuxhint.com/centos/>

In another case, you need config “become\_ask\_pass=True”, and you need to type in the password during the deployment when the user sudo(and you just need to type in the password for one time during the deployment).

# Online mode

Almost nothing need to do now.

cd kubespray dir, then type the command:

(base)[:)@vm kubespary]# conda activate env\_1

(env\_1)[:)@vm kubespary]# ansible-playbook -i inventory/mycluster/hosts.yaml \

-b --become-user=root \

-e "ansible\_user=username ansible\_ssh\_pass=password" \

cluster.yml

Remember to change below params:

username: The same in “ssh username@IP”

password: The password use ssh to connect to IP

# Offline mode

Kubespray also provides scripts for you to deploy k8s in offline mode.

It’s not truly offline, but the target node doesn’t have the access to the url to download related files and pull images.

## README.md provided by Kubespray

[:)@vm path]# cd kubespray/contrib/offline

Here are the docs you may want to check:

- README.md in current dir;

- <https://blog.k8s.li/kubespray-tips.html#%E4%BA%8C%E8%BF%9B%E5%88%B6%E6%96%87%E4%BB%B6>

As the README.md suggests, you must have one VM which can download all the images you need, and you need then deploy the images to the local registry of the VM you what to set up k8s.

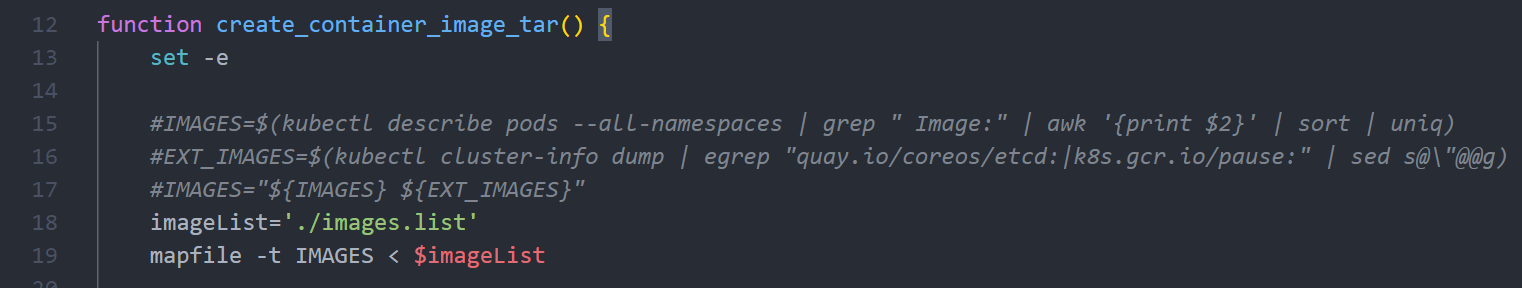
## Problems may occur (continually updated)

In most cases, the BKM is following the README.md, but you may come across several problems during the setup.

### Version Confliction of the images

This problem will occur when the version of images in images.list is not the same as the images you download.

You can adjust the function named “create\_container\_image\_tar” in **manage-offline-container-images.sh** to solve the version confliction problem as shown below.

Just replace the action getting images’ information from an existed k8s node of getting images’ information from the images.list, and make sure that the images.list has all the images you need.

The code may changed cause kubespray will be updated day by day.

The guiding ideology to solve the problem of version confliction is using scripts to get consistent images you need.